2000-06

A business process redesign of the U.S. Coast Guard Port State Control boarding process

Fosdick, Jason A.

Monterey, California. Naval Postgraduate School

http://handle.dtic.mil/100.2/ADA380920

NAVAL POSTGRADUATE SCHOOL
Monterey, California

THESIS

A BUSINESS PROCESS REDESIGN OF THE U. S. COAST GUARD PORT STATE CONTROL BOARDING PROCESS

by

Jason A. Fosdick

June 2000

Thesis Advisors: Daniel R. Dolk
Mark E. Nissen

Approved for public release; distribution is unlimited.

REPORT DOCUMENTATION PAGE

Form Approved OMB No. 0704-0188

1. AGENCY USE ONLY ***(Leave blank)***	**2. REPORT DATE** June 2000	**3. REPORT TYPE AND DATES COVERED** Master's Thesis
4. TITLE AND SUBTITLE : A Business Process Redesign of the U. S. Coast Guard Port State Control Boarding Process		**5. FUNDING NUMBERS**
6. AUTHOR(S) Fosdick, Jason A.		
7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES) Naval Postgraduate School Monterey, CA 93943-5000		**8. PERFORMING ORGANIZATION REPORT NUMBER**
9. SPONSORING / MONITORING AGENCY NAME(S) AND ADDRESS(ES) N/A		**10. SPONSORING / MONITORING AGENCY REPORT NUMBER**

11. SUPPLEMENTARY NOTES

The views expressed in this thesis are those of the author and do not reflect the official policy or position of the Department of Defense or the U.S. Government.

12a. DISTRIBUTION / AVAILABILITY STATEMENT Approved for public release; distribution is unlimited.	**12b. DISTRIBUTION CODE**

13. ABSTRACT ***(maximum 200 words)***

The United States Coast Guard Port State Control (PSC) is a port entry tracking process, which is currently performed primarily using paper and pencil. This thesis examines the feasibility and effectiveness of redesigning the PSC process in light of modern Business Process Redesign methodologies that incorporate contemporary information technology. The current process is modeled using the automated redesign tool, KOPeR, to identify pertinent redesign recommendations. A redesign of the process is completed using the recommendations provided by KOPeR and leveraging existing Coast Guard infrastructure and technology solutions. The effectiveness of the redesigned process is evaluated against the current process by using discrete event simulation models to compute the relative cycle times. Three different scenarios are run which show a potential annual reduction in manpower ranging from two to four person years. A Web-based prototype system, Re-engineered Port System (RePortS), is developed using basic tools such as Microsoft Access and Active Server Pages to demonstrate the feasibility of implementing the required functionality. The benefits of replacing the current manual system with a Web-based system are, reduced cycle time, increased accuracy and consistency in the process.

14. SUBJECT TERMS U.S. Coast Guard, Business Process Reengineering, Database, Web-Database Connectivity	**15. NUMBER OF PAGES** 168
	16. PRICE CODE

17. SECURITY CLASSIFICATION OF REPORT Unclassified	**18. SECURITY CLASSIFICATION OF THIS PAGE** Unclassified	**19. SECURITY CLASSIFICATION OF ABSTRACT** Unclassified	**20. LIMITATION OF ABSTRACT** UL

NSN 7540-01-280-5500

Standard Form 298 (Rev. 2-89)
Prescribed by ANSI Std. 239-18

THIS PAGE INTENTIONALLY LEFT BLANK

Approved for public release; distribution is unlimited

A BUSINESS PROCESS REDESIGN OF THE U. S. COAST GUARD PORT STATE CONTROL BOARDING PROCESS

Jason A. Fosdick
Lieutenant Commander, United States Coast Guard
B.S., United States Coast Guard Academy, 1988

Submitted in partial fulfillment of the
requirements for the degree of

MASTER OF SCIENCE IN INFORMATION TECHNOLOGY MANAGEMENT

from the

NAVAL POSTGRADUATE SCHOOL
June 2000

Author: [redacted]
Jason A. Fosdick

Approved by: [redacted]
Daniel R. Dolk, Thesis Advisor

[redacted]
Mark E. Nissen, Associate Advisor

[redacted]
Dan Boger, Chairman
Department of Computer Science and
Information Systems Academic Group

THIS PAGE INTENTIONALLY LEFT BLANK

ABSTRACT

The United States Coast Guard Port State Control (PSC) is a port entry tracking process, which is currently performed primarily using paper and pencil. This thesis examines the feasibility and effectiveness of redesigning the PSC process in light of modern Business Process Redesign methodologies that incorporate contemporary information technology. The current process is modeled using the automated redesign tool, KOPeR, to identify pertinent redesign recommendations. A redesign of the process is completed using the recommendations provided by KOPeR and leveraging existing Coast Guard infrastructure and technology solutions. The effectiveness of the redesigned process is evaluated against the current process by using discrete event simulation models to compute the relative cycle times. Three different scenarios are run which show a potential annual reduction in manpower ranging from two to four person years. A Web-based prototype system, Re-engineered Port System (RePortS), is developed using basic tools such as Microsoft Access and Active Server Pages to demonstrate the feasibility of implementing the required functionality. The benefits of replacing the current manual system with a Web-based system are, reduced cycle time, increased accuracy and consistency in the process.

THIS PAGE INTENTIONALLY LEFT BLANK

TABLE OF CONTENTS

LIST OF FIGURES

THIS PAGE INTENTIONALLY LEFT BLANK

LIST OF TABLES

LIST OF ACRONYMS AND ABBREVIATIONS

ADO	ActiveX Data Objects
ASP	Active Server Pages
BPR	Business Process Reengineering
CGSWIII	Coast Guard Standard Workstation III
CGWEB	Coast Guard Intranet
COTP	Captain of the Port
DFD	Data Flow Diagram
DSN	Data Source Name
FIFO	First In First Out
FV	Foreign Vessel
HTML	Hyper Text Markup Language
HTTP	Hyper Text Transfer Protocol
IIS	Internet Information Server
MSIS	Marine Safety Information System
MSM	Marine Safety Manual
MSN	Marine Safety Network
MSO	Marine Safety Office
OCMI	Officer in Charge Marine Inspection
ODBC	Open Database Connectivity
PSC	Port State Control
RAD	Rapid Application Development
SOM	Semantic Object Model
SQL	Structured Query Language
SSL	Secure Sockets Layer
USCG	United States Coast Guard
WWW	World Wide Web

THIS PAGE INTENTIONALLY LEFT BLANK

ACKNOWLEDGEMENTS/DEDICATION

My sincerest thanks and gratitude go to my thesis advisors, Professor Daniel Dolk and Professor Mark Nissen. They provided a steady guiding hand in the development of this thesis. I thank them for the time they spent making this a success.

I dedicate this work to my wife, Kristine who sat by me day in and day out and provided invaluable support, encouragement and expertise in making the completion of this thesis a success.

THIS PAGE INTENTIONALLY LEFT BLANK

I. INTRODUCTION

A. BACKGROUND

The United States Coast Guard (USCG) is charged by Congress to ensure that foreign flagged vessels entering US ports are adhering to US laws and International treaties. To complete this mission, the Coast Guard has created the Port State Control (PSC) program to board foreign vessels to ensure compliance. The current process of gathering vessel port call information, deciding which vessels to board, boarding them, and then documenting the boarding is largely completed using paper and pencil. This leads to a process that takes considerable time and leads to multiple errors and rework due to the number of times data must be manually transferred. Such a process is an ideal Business Process Reengineering (BPR) candidate because of the opportunity to reduce cycle time and data collection errors.

The Coast Guard recently initiated a BPR of the PSC process in conjunction with a large software project to replace a legacy vessel database system running on Prime computers. The BPR was performed by personnel at Coast Guard Headquarters interviewing field units experienced in performing the tasks involved in the PSC process. From this BPR, a set of use cases was developed to assist the contractor in writing the software to implement the new process. Unfortunately, the contract was terminated, and replaced by a much smaller project undertaken whose scope was simply to replace the functionality of the legacy database with a database running on a modern platform. The additional functionality associated with the BPR for the PSC process and other areas of functionality are planned as incremental changes to the new database over a period of time. This provides an opportunity to perform an independent BPR using current methods in order to provide alternative solutions for the eventual implementation of the redesign.

The main purpose of this thesis is to analyze and perform a business process redesign (BPR) of the U. S. Coast Guard Port State Control (PSC) process. The focus of this research is to provide innovative solutions that dramatically improve the cycle time and data accuracy of the process.

B. PURPOSE

This research examines the U.S. Coast Guard's Port State Control process at the field unit level. The objective is to significantly improve the critical measures of performance, in terms of cycle time and data integrity, by redesigning the current process.

Presently, the Coast Guard is designing a new enterprise database application, called the Marine Safety Network (MSN), which will support parts of the PSC process. While early versions of the MSN will not include the types of functionality presented in this research, it is anticipated that future releases of the MSN can implement features discussed in this study. Further, the redesign methods outlined in this thesis can be applied to other similar marine safety processes where similar improvement can be realized.

C. SCOPE AND METHODOLOGY

This study focuses on the state and the inefficiencies of the current PSC process and ways to eliminate or reduce the effects of these inefficiencies. The primary objective is to define the process and perform a redesign that improves performance.

The process under study consists of an administrative activities portion and a physical inspection of a vessel. This thesis covers only the administrative portion of the process. The prototype developed for the redesign process will consist only of those elements needed to implement a working demonstration of the redesigned process. It will therefore not include encryption/security features, full implementation of all features of the redesign and any field testing of the prototype.

The methodology followed in fulfilling the objective consists of several steps:

1. Conduct a literature search of current BPR methodologies and associated technologies, and select a BPR methodology for the redesign of the process.
2. Conduct a review of Coast Guard instructions and directives pertaining to the PSC process.
3. Identify and model the current state of the PSC process.

4. Identify and analyze the pathologies and problems with the current PSC process.
5. Redesign the PSC process to improve process cycle time and data collection accuracy.
6. Create simulation models of the current and redesigned processes to assess the effectiveness of the redesigned process.
7. Create a prototype/proof of concept of the redesigned process using web enabled database technology.

D. ORGANIZATION OF STUDY

The thesis is organized as follows. Chapter II provides an overview of the PSC process and BPR. Chapter III contains a logical breakdown of the process and models of the sub-processes using simple diagrams and the diagnosis of the processes. Chapter IV contains the proposed process redesigns and simulation models of the proposed and current processes. Chapter V describes the software prototype architecture and implementation. Chapter VI summarizes the conclusions and recommendations.

THIS PAGE INTENTIONALLY LEFT BLANK

II. PROCESS OVERVIEW

A. THE PORT STATE CONTROL PROCESS

To understand the genesis of the current PSC process, it is necessary to review the organizational structure and history of the US Coast Guard Marine Safety program. The Coast Guard has established Marine Safety Offices (MSO) in the major port areas in the US to carry out the missions of the Marine Safety program.

Prior to the establishment of MSOs, the Marine Safety program was administered by two separate commands in the major port areas. These commands were the Captain of the Port (COTP) and the Officer in Charge, Marine Inspection (OCMI) respectively. The major responsibilities of the COTP were port security, port safety, and marine environmental response/prevention. The OCMI was responsible for the inspection of US flagged vessels, investigating marine casualties and the licensing of US Merchant Mariners. Due to budgetary pressures and the economies of having a single command versus two separate commands, it was decided to merge the two different commands into one with the Commanding Officer performing both roles of COTP and OCMI. The resulting command was called an MSO.

The current Coast Guard PSC process has its roots in a precursor program called the Foreign Vessel (FV) boarding program. The goal of the FV program was to ensure foreign registered vessels were complying with US laws and regulations while in US waters. The program at most MSOs was carried out by the Port Operations Department, which is the COTP arm of the MSO. The boardings of the vessels were carried out by Petty Officers who usually had a couple of months of on the job training, specific rate experience, and time spent at a Marine Safety "C" school for program familiarity. This program worked well in enforcing the pollution prevention and navigation safety regulations on the foreign vessels calling at US ports.

With the increase in global competition and increasing costs associated with operating a US registered vessel, the US deep draft vessel fleet was on the decline. By the mid to late 1980's the majority of vessels calling on US ports were foreign registered with the majority being registered in a country with low registration costs and less stringent safety standards. The monetary rewards of registering a vessel in one of these "flags of convenience" were substantial and the registries of these countries grew dramatically.

In 1993 there was a series of incidents, the most notable being a hazardous materials spill off of the New Jersey coast, involving foreign registered vessels in US waters that had been recently boarded by different MSOs. These incidents caught the attention of Congress, which held hearings on the matter. It was found that there needed to be a more in depth inspection of the foreign vessels calling on US ports and that these inspections needed to be carried out by more experienced personnel. In the 1994 Department of Transportation Appropriations Bill, the Congress mandated that the Coast Guard change its approach to foreign vessel boardings to "hold those most responsible for substandard ships accountable, including owners, classification societies and flag states." (U. S. Coast Guard Marine Safety Manual Vol. II, Chap 23) Thus, the current PSC program was born.

The organizational difficulties of persuading the Port Operations Department and the Vessel Inspections Department (the OCMI arm of the MSO) to agree initially on who should control the program were problematic. Nevertheless, differences were ironed out and workable solutions to the organizational structure were generated independently in each port area. This resulted in each MSO performing the PSC program differently and interpreting guidance as they saw fit, which resulted in the program being applied inconsistently nationwide. Recognizing this, Coast Guard Headquarters published guidance and requirements on how to perform the program to meet key goals.

1. Description of the Current Process

The majority of the mechanics of the PSC program and its associated processes is spelled out to field units in the USCG Marine Safety Manual, Volume II, Chapters 19 through 24. There are other vessel inspection oriented instructions relating to the PSC program, but they are not germane to the administrative processes that constitute the focus of this research.

In order for field units to successfully execute the PSC program, a process must be followed which allows the unit to identify vessels requiring boardings, perform the boarding and then document the boarding. This process is the focus of this thesis.

Figure 2.1 depicts a general overview of the PSC process. A short explanation of each step is provided for clarification and a full explanation of each step will follow in Chapter 3.

1. The process begins with a vessel agent calling either a Coast Guard point of contact or a centralized broker of vessel arrivals to report the arrival of a vessel.
2. The pertinent data on the vessel is recorded on a log sheet and held until the PSC section personnel gather the log sheets periodically through out the day.
3. One of the personnel in the PSC section enters the vessel arrival information from the log sheets into the Coast Guard Marine Safety Information System (MSIS) and prints a history of each vessel's previous port calls and boardings throughout the US.
4. Based on information from the history of the vessel, a grading sheet is prepared to determine the priority of the vessel for boarding.
5. The resulting information, vessel name and priority are entered into another log based on the arrival date of the vessel.
6. A supervisor then reviews the completed grading sheets and histories.
7. Based on vessel priorities, dates of arrival, and number of personnel available, boarding decisions are made.
8. Vessel boarding teams are then dispatched.
9. On board the vessel, the boarding team verifies the various information about the vessel against the history pulled from MSIS. Changes to the information and the results of the boarding itself are recorded in a "boarding book" as documentation of the boarding.

10. When the boarding is complete, the boarding team leaves a hand written boarding letter with the Master of the vessel detailing the results of the exam and listing discrepancies found, if any.
11. Once back at the office, the boarding team then prepares the documentation of the vessel boarding. This entails entering the hand written changes to vessel information, the results of the boarding, and any discrepancies into MSIS.
12. Once the data entry is complete, all information changed in MSIS is printed out for inclusion in the local vessel paper file.
13. Both the print outs and the vessel "boarding book" are submitted for review.
14. After the supervisor has reviewed and approved the boarding package, it is filed with all vessel-boarding documents in a local filing system.
15. For vessels not boarded, an entry is made in MSIS to indicate the vessel made a port call, but was not boarded.

2. The Goals of the PSC Process

The overarching goal of the PSC program is to eliminate substandard vessels from US waters (U. S. Coast Guard Marine Safety Manual, Vol. II). The subgoals for the PSC process are efficiency, timeliness and accuracy in the following areas:

1. Identification of vessels requiring boarding.
2. Gathering of data on the current state of a vessel being boarded.
3. Documentation of vessel boardings.

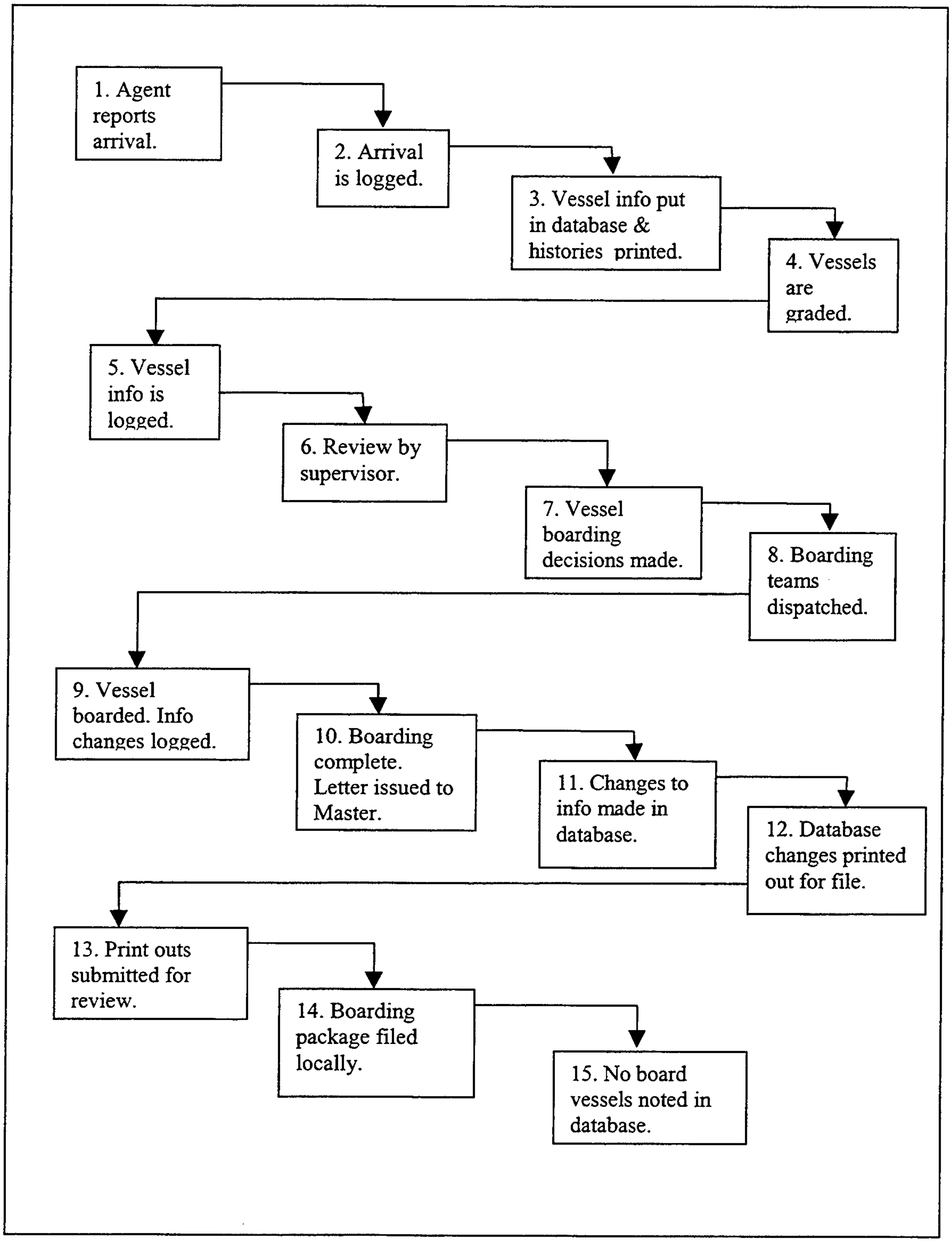


Figure 2-1. The PSC Process

3. Critical Performance Measures

In order for the PSC process to perform optimally, the subgoals outlined in the preceding section must be met. If they are not, time and resources will be wasted, not only by the Coast Guard, but also by the vessel inconvenienced with an unnecessary boarding.

Based on the subgoals it is clear that cycle time and data accuracy are the critical performance measures for this process. Cycle time is the time required to complete the process, from the time the vessel agent calls in the vessel arrival, to the completion of the documentation of the vessel boarding. Cycle time is measured for each subprocess as described in Chapter III. Data accuracy/redundancy is measured by the number of times identical data has to be copied from one piece of paper to another or input into a database. The data accuracy measure is taken for each subprocess.

B. BUSINESS PROCESS REENGINEERING

Business Process Reengineering has occupied a dominant spot in the IT landscape throughout the 1990s. Due to the amount of attention given to the BPR movement many different BPR methodologies have evolved ranging from extreme to very mild with respect to the degree of changes to processes and expectations of improvement. This section provides a brief introduction to BPR, a look at several different BPR approaches, a brief introduction to the chosen redesign methodology, and an evaluation of the applicability of the chosen methodology to the PSC process.

1. Reengineering Overview

When introduced by Davenport and Short (1990) and Hammer (1990), the term *business process reengineering* applied to a radical and far-reaching overhaul of a business process. Since that time the term has become more generic and now includes a broader mix of methods for process redesign, which range from the original meaning to simple process improvement techniques. (Baden and Peters, 1997, p51) While there are many different reengineering methodologies, these methodologies can be categorized into one of three different approaches: Continuous Process Improvement, Business Process Redesign and Business Process Reengineering. While each approach is different in the amount of change it tries to effect, the goal remains the same, namely to change the way the business is organized to perform its work processes. "Most reengineering methodologies investigate ways to eliminate non-value added processes, utilize information technology to minimize redundant data entry and storage, integrate or combine similar processes, implement data sharing, and automate manual processes." (Baden and Peters, 1997, p51) Table 2-1 compares the major features of the three approaches.

2. Business Process Reengineering Approaches

Before selecting a specific methodology to use in the redesign of the PSC process, it is important to understand where in the spectrum of approaches a redesign lies. Additionally, it is important to understand the crucial differences between the three different approaches.

Business Process Reengineering is "the fundamental rethinking and radical redesign of business processes to achieve dramatic improvements in critical, contemporary measures of performance, such as cost, quality, service and speed." (Hammer and Champy, 1993, p32) The basic premise is to take a process, identify the desired outcome, and build a new process "rejecting the conventional wisdom and received assumptions of the past." (Hammer and Champy, 1993, p49) The result is a process that provides a quantum leap in performance and may change the entire structure of the organization. This approach is not without its risks because of the vast changes it endorses and the potential costs associated with implementing a radically new process.

Features	Continuous Process Improvement	Business Process Redesign	Business Process Reengineering
Philosophy	Improve what you do in functional or sub-activity; Accepts status quo – current processes are what customers need	Accepts current process: Remove "hand off" activities of little value in an end-to-end examination	Focus on critical broken processes: Alter or replace basic approach to doing business in jobs, skills, structures, systems, culture
Timing	Part of a way of life to continuously improve; project results in short time frames	Done on a periodic basis; improvement may take a few months for simple efforts; 1 to 2 years if efforts are more complex	Used selectively; sub-process deployment may take several months; full deployment across an entire complex process may take 2 to 5 years
Scope	Little emphasis on interrelationship of business processes in a business system; internal focus	Coverage of many sub-processes and "turf"; internal focus	Scope is entire process or major sub-processes that cover broad cross-functional areas; includes interfacing outside the organization
Leadership	Broad-based, bottom-up	Both bottom-up and top-down, more senior leadership needed	Management focused, top-down; significant senior management attention and time
Means	Generally, improvement work done by work unit part-time teams; use of quality tools	Improvement work often done by diversified task forces or teams that cross functions	Improvement generally done by dedicated teams representing end-to-end activities; work facilitated by process sponsors and owners
Performance Gains	Incremental: Slightly increases (5-10%) performance	Moderately increases performance	Revolutionary: Greatly increases performance
Costs, Risks, Pain	Low: Resources generally easily handled within existing budgets and personnel allocations; small iterative investments; low-level effort offers few risks; pain of implementation is minimal	Low to moderate: Resources may require shifting funds and personnel or adding more funds and personnel; risks increase somewhat as more activities are involved; implementation pain covers more activities	High: Resources require significant funding and dedicated personnel allocations; large, upfront investment; risks greatly increase given extensive process coverage; implementation pain is high

Table 2-1. Process Improvement Approaches. From Caudle, 1995

Business process redesign can be viewed as either a subset of a larger business process reengineering effort or may be a project involving a single process in an organization. The expectations from a process redesign are lower than those associated with a reengineering effort, as the changes to the organization are not as great, and the structure of the process is generally accepted in its current state. The efforts of a business process redesign usually focus on removing non-value added activities and reducing the number of personnel needed to perform the process by either leveraging technology, or integrating tasks in a process. (Caudle, 1995)

Continuous process improvement has its roots in the Total Quality Management movement. The idea is to constantly improve the process with incremental changes suggested by those performing the process. This approach is decidedly low risk, as the types of changes made to the processes usually cost little to implement and do not make 'revolutionary' changes to the process or organization. (Caudle, 1995) This approach could arguably not qualify as reengineering, especially in the eyes of Hammer and Champy, but is included nevertheless since it does focus on improving a business process.

The redesign of the PSC process addressed in this thesis qualifies most closely as the intermediate approach (i.e., business process redesign) with some characteristics of the reengineering approach. Since law mandates the PSC program, the process cannot be totally redesigned from a "clean sheet" as espoused by Hammer and Champy. Rather, the current process will have to be accepted as the process to follow to reach the end goal of boarding foreign vessels. This does not mean, however, that the organizational structure of the process cannot be adjusted, or that tasks cannot be integrated. It is expected that the use of technology in the redesigned process will be a key enabler to implementation of the process.

The methodology employed for the PSC process redesign is one introduced by Mark E. Nissen in "Redesigning Reengineering through Measurement-Driven Inference". This methodology is a "blend of expert reengineering methodologies" (Nissen, 1998, p511) and uses Knowledge Based Systems technology to automate part of the redesign process. The general redesign process is depicted in Figure 2-1.

The first step is to identify a process for redesign. Next, the process is modeled using nodes, directed edges and process attributes to facilitate measurement.

Measurements of the process are then taken, using the process measures shown in Table 2-2. From the measurements taken, specific "pathologies" are identified, and then used to match appropriate redesign transformations for the process redesign. The process is then redesigned using the transformations identified in the previous step, usually with more than one redesign candidate generated. Once redesigned, the processes are tested, often with simulation, and finally a "preferred" process is identified for implementation. (Nissen, 1998)

For this research, the process of diagnosing pathologies and matching transformations is performed using KOPeR, the "proof-of-concept Knowledge Based System". (Nissen, 1998) This gives the redesign the benefit of the collective knowledge gained by almost a decade of BPR.

Simulation of the current and redesigned processes is carried out using simulation models built in the simulation software package, EXTEND+BPR®. Parameters for the simulation models are based on my extensive experience of supervising the PSC program at MSO/Group Los Angeles – Long Beach, California, one of the nations largest and busiest port complexes.

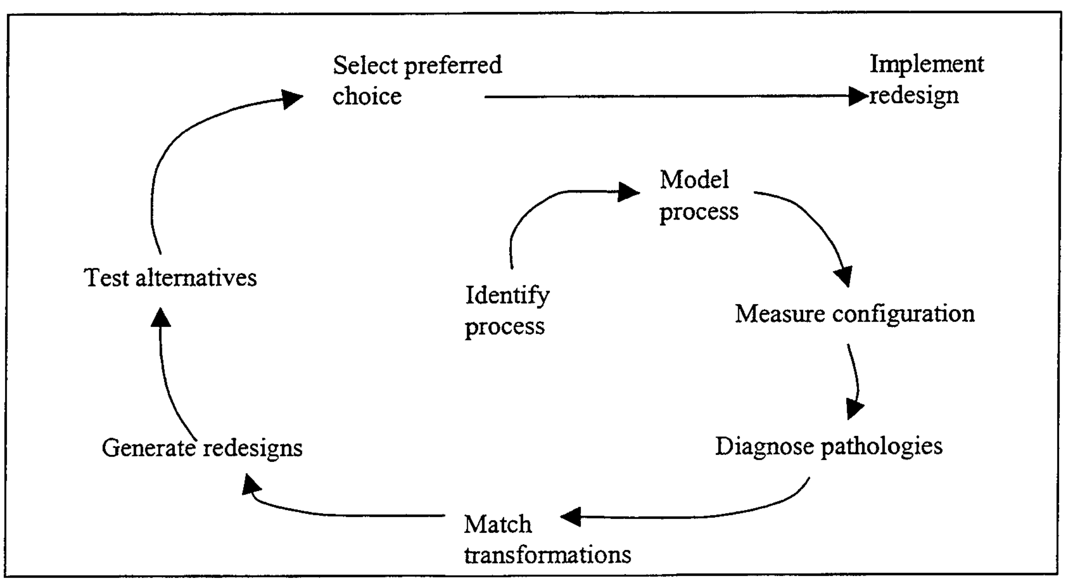


Figure 2-2. From Nissen

Measure	Graph-Based Definition
Process Length	Number of nodes in longest path
Process Breadth	Number of distinct paths
Process Depth	Number of process levels
Process Size	Number of nodes in process model
Process Feedback	Number of cycles in graph
Parallelism	Process Size divided by Length
IT Support	Number of IT-support attributes
IT Communication	Number of IT-communication attributes
IT Automation	Number of IT-automation attributes
Organizational Roles	Number of unique agent role attributes
Process Handoffs	Number of inter-role edges
Organizations	Number of unique agent organizational attributes
Value Chains	Number of unique activity Value Chain attributes

Table 2-2. From Nissen

3. Applicability to the Port State Control Process

The PSC process is a suitable candidate for a business process redesign. It consists of a multitude of tasks performed by several different personnel with several redundant steps visible to even those unfamiliar with the entire process. The process is well defined and is highly repeatable; two important attributes for performing the redesign methodology as described by Nissen.

As with many of the processes performed throughout the Coast Guard, the PSC process operates with little information technology support. The information technology support used by the process is a legacy database running on antiquated hardware. The timing is right to leverage the Coast Guard's investment in new information technology infrastructure by redesigning processes to take advantage of this technology.

Additionally, the redesign of the PSC process is consistent with the US Coast Guard Information Technology Management Strategy vision which states: "The Coast Guard, as the world's premier maritime service, delivers the right information to the right people at the right time to support all Coast Guard Missions." (US Coast Guard Information Technology Management Strategy, 1998)

THIS PAGE INTENTIONALLY LEFT BLANK

III. THE PSC PROCESS MODEL

A. INTRODUCTION

This chapter presents the model of the current PSC process. The first step is to decompose the process into major sub processes, which allows for finer granularity in taking measurements and a modular approach to the implementation of the redesigned process. A graphical process model is created for each sub process as well as a simulation model. The simulation models yield baseline process cycle time measurements and are covered in depth in Chapter IV. Measurements are taken from the graphical model and then run through KOPeR, which helps identify transformations for each sub process. The meaning and impacts of the measurements and transformations are discussed. Then based on these transformations potential improvements and benefits are identified.

The current process can be decomposed into two logical segments that are distinct in their purposes and outcomes: the targeting process and the vessel boarding process. The targeting process starts at the beginning of the PSC process, proceeds up to, and includes, the assignment of boarding teams to the selected vessels. The boarding process picks up where the targeting process leaves off, and encompasses the remainder of the PSC process.

B. TARGETING PROCESS MODEL

This section discusses the targeting process model. The process model is described, with an eye toward identifying activity inputs and outputs, and depicted using a graphical method. Improvements and benefits to the process are then discussed later in this section. Figure 3-1 is a graphical depiction of the current targeting process. The activities presented in Figure 3-1 are used to guide discussion of the current targeting process. Each activity is identified by a node number and is connected to the next node with a directed edge. The node number as well as the activity name are identified to provide clarity in the discussion of the activity.

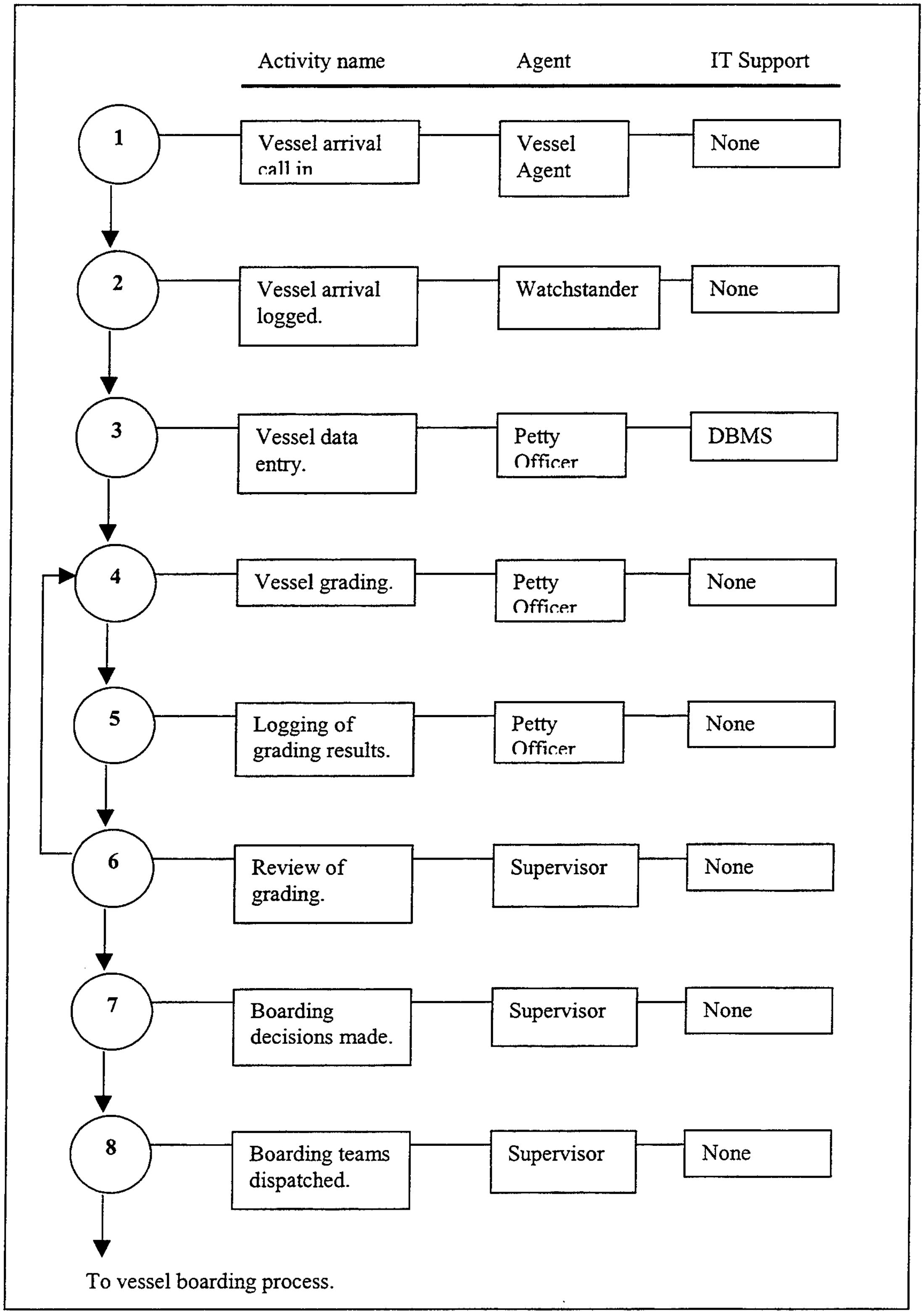


Figure 3-1. PSC Targeting Process

1. Current Process Model

The targeting process starts with activity *node one,* vessel arrival call in. A vessel agent calling in a vessel arrival to a Coast Guard watchstander accomplishes this activity. The agent provides the name of the vessel, the berth it will be occupying, the date of arrival, the official number of the vessel, the vessel's cargo, and the agent's name and agency.

Activity *Node two*, vessel arrival logged, is the next step. In this activity, the watchstander logs the information provided by the agent. The PSC section personnel collect the log twice a day, once in the morning and once at noon.

Activity *node three*, vessel data entry, one person in the PSC section is delegated to enter data from the log sheets into the Coast Guard Marine Safety Information System (MSIS). The user inputs the information from the vessel arrival log into a computerized form. When the data is submitted, a product called the vessel history is queued up to print and a unique case number is created for each vessel input to the system. The vessel history contains the pertinent data on the vessel as well as all Coast Guard contacts on the vessel.

Activity *node four*, vessel grading, the grade a vessel receives is based on its history using a paper matrix. A copy of the matrix, from the USCG Marine Safety Manual (MSM) Chapter 20, is included as Appendix A to this work. The matrix is annotated with the name of the vessel, official number, date of arrival, berth, and the results of the grading. The grade a vessel receives is based upon the number of points the vessel scores on the criteria supplied on the matrix. Based upon the number of points scored on the matrix, the vessel is assigned a priority between one and four, with one being the highest priority.

After the grading is complete, activity *node five*, the logging of grading results, occurs. The person who performed the grading compiles the vessel names and priorities into a log that houses vessel targeting information and names. The grading sheets and histories are filed by the date the vessels are due to arrive.

This leads into activity *node six*, review of grading. A supervisor then reviews each grading sheet and history for errors and makes any changes needed to the log and the histories. If the quality of the grading sheets is not up to the supervisor's expectations, feed back is provided to the person who performed the activity at node four. The supervisor also is empowered to downgrade or upgrade the priority on a vessel during the review based on a set of rules provided in the MSM.

Once the supervisor makes the adjustments, activity *node seven*, boarding decisions made, begins. Based on the priority, each vessel is considered for a boarding with priority one vessels always being boarded and priority four vessels never being boarded.

After the supervisor has decided on the vessels to board, activity *node eight*, boarding teams dispatched, is the final node to occur. The supervisor prioritizes the vessels targeted for boarding based on the number of personnel available for the day and assigns boarding teams to each vessel. Each boarding team is provided the vessel history and grading sheet for use in the vessel boarding process.

The depiction of the PSC targeting process in Figure 3-1, in addition to providing a template for discussion, is also suitable for taking measurements for input into KOPeR. The attributes of each node (e.g., process name, the entity performing the process, and the IT resource used for support) describe the pertinent measurement items present at the node. The measurements for the process that KOPeR needs in order to complete a redesign recommendation are shown in Table 3-1. (Nissen 1998)

Configuration Measure	**Value**
Process Size	8
Process Length	8
Handoffs	3
Feedback loops	1
IT-Support	1
IT-Communication	0
IT-Automation	0

Table 3-1. Measurements for the PSC Targeting Process

Before continuing, it is important to understand the definitions of the measures of the process. Process size is the total number of nodes in the process model, in this case all of the circles in the graphical model. Process length is the number of nodes in the longest path in the process. Handoffs are the number of times the agent role changes to a different role. Feedback loops are the number of cycles from one node to another in the opposite direction of the flow in the graph. IT-support is the number of IT-support attributes. IT-communication is the number of IT-communication attributes. IT-automation is the number of IT-automation attributes. (Nissen 1998, p.513)

These measures are used by KOPeR to diagnose the pathologies of the process and then give recommended transformations based on those pathologies. KOPeR, being a "proof of concept" system, is obviously limited in the types of pathologies and recommendations it can give. Therefore, it will be necessary to provide additional manual analysis of the process to arrive at recommendations with suitable detail to perform a redesign of the process.

2. Possible Ways to Improve the Process

Giving the process measurements in Table 3-1 to KOPeR results in a diagnosis and a list of recommendations. The diagnosis provides the pathologies that were detected from the measurements. The diagnostic output from KOPeR is provided in Table 3-2. To perform this diagnosis, KOPeR has to transform the measurements given into a fraction, which allows improved interprocess comparability and makes the system more robust to variability in process sizes. These fractions are arrived at by dividing the process size into the particular measure (example, size = 8, handoffs = 3, 3/8 = 0.375). (Nissen 1998, p.531)

KOPeR Diagnosis for the Targeting Process
Measurements (e.g., size of 8) suggest the *small PSC Targeting process* suffers from the following pathologies: • Parallelism (1.0) - *sequential process.* • Handoffs fraction (0.375) - *process friction.* • Feedback fraction (0.125) - *feedback looks OK.* • IT support fraction (0.125) - *inadequate IT support.* • IT communication fraction (0.0) - *inadequate IT communications.* • IT automation fraction (0.0) - *IT automation first requires substantial infrastructure in terms of support and communication.*

Table 3-2. KOPeR Diagnosis (From KOPeR Web Page)

To better understand the diagnosis and the recommendations that derive from them, it is helpful to understand what each of the items in Table 3-2 means and the performance implications that pertain to them.

Parallelism is a measure of how linear or sequential the process is, with a measure of 1.00 being the minimum value for this measure. It is arrived at by dividing the process size by the process length. The diagnosis for this measure provided by KOPeR is "sequential process". This is based solely on KOPeR knowing that a parallelism value of 1.00 is the minimum value for the measure. As stated by Nissen, benchmarking in the domain must be employed to determine if the specific level of parallelism is pathological. (Nissen 1998, p.516) In the case of the PSC targeting process, it is logical to infer that a parallelism measure of 1.00 (the minimum) is pathological and that the process does in fact suffer from the diagnosis of "sequential process".

The handoffs fraction is a measure of the amount of job specialization and the number of times the process is fragmented into smaller pieces. For the targeting process, KOPeR has given the pathology of "process friction". This equates to an amount of friction in the process and the related delays or increases in cycle time in the process. The fraction is computed by dividing the handoffs measure by the size of the process.

The implications of a handoffs fraction greater than zero are attendant delays in the system due to inefficiencies in the way the work flow is laid out or the way it is completed

The feedback fraction is a measure of the amount of reviews in the process. For the targeting process, KOPeR has given an "OK" for the pathology. The fraction is calculated by dividing the feedback measure by the process size. This measure is one that would indicate lost time in the process due to excessive reviews, not empowering personnel to make decisions, or a centralized control figure in the process. This is another area that can increase cycle time in a process.

The IT support fraction is a measure of the amount of information technology support in the process. It is calculated by dividing the IT-support measure by the process size. For the targeting process, KOPeR gave a pathology of "inadequate IT-support". This would point to a process where all of the processes are done manually, and there is little to no use of modern information technology tools. This measure also takes in part of the IT infrastructure provided to support the process, the other part of the infrastructure is covered under IT communication.

The IT communication fraction is a measure of the use of information technology to provide inter activity communication in the process. For the targeting process, KOPeR provided a pathology of "inadequate IT communications". This measure, like the others, is calculated by dividing the IT-communications measure by the process size. Using more information technology to assist with inter activity communication would speed the process and lower process cycle time. Examples of IT-communication would be email, shared databases or workflow systems.

The IT-automation fraction is a measure of the usage of information technology to provide automation within the process. The pathology provided by KOPeR, for the targeting process, was that "IT automation first requires substantial infrastructure in terms of support and communications". While cryptic at first, it does make sense when looking at the pathologies identified for IT communications and support. If there is little in the way of IT communications or support it follows naturally that automation is a ways down the road. Automation would involve the use of IT support and communications to further automate activities in the process. Automation could take the form of intelligent software agents performing the process on their own or could be as simple as a piece of code that transforms data input to a usable output.

KOPeR then provides a set of recommendations for transformations to the process based on the pathologies identified above. These recommendations are contained in Table 3-3.

The recommendations from KOPeR are generic and require the application of specifics to complete the process redesign. Examination of the process in light of each recommendation will bring out some ways of redesigning the process to improve performance.

KOPeR Recommendations for the Targeting Process
De-linearize process activities to increase parallelism; such activities must be sequentially independent (e.g., have mutually exclusive inputs and outputs).
Try a case manager or case team to decrease friction; be sure to include a source of expertise.
Look to information technology to increase support to process activities; decision support systems and desktop office tools generally have good payoffs and intelligent systems can greatly enhance knowledge work; be sure to address personnel training and maintenance of the IT.
Look to information technology to increase support to process communications; e-mail and shared databases through local/wide area networks generally have good payoffs and workflow systems can greatly expedite process flows; be sure to address personnel training and maintenance of the IT.
Look to information technology to automate process activities, but note that substantial IT infrastructure is first required, particularly in terms of process support and communication; try workflow systems for support and communication, and then look to intelligent agents, which can enable many electronic commerce opportunities.
In addition to delinearization and the use of a case manager, workflow systems offer good potential for process improvement; this requires substantial IT infrastructure and support however.

Table 3-3. KOPeR Recommendations (Output from KOPeR Web Page)

Delinearizing the process may be a viable alternative; but each process activity is not sequentially independent. To illustrate this the process will be examined by following one vessel arrival from the beginning to the end of the process. Activity node numbers are taken from Figure 3-1. All nodes are contingent upon the previous node when addressing one vessel in the system. When there are multiple vessels in the system, certain nodes can be completed concurrently for different vessels in the system (e.g., node four and node six activities can be completed at the same time but on different vessels). At *node one*, the vessel arrival is called in to *node two* where it is logged into the arrival log. *Node three* requires the information from node two to perform the task of vessel data entry to MSIS. *Node four* requires the vessel history from MSIS to grade the vessel. *Node five* requires the results from node four to log the results of the grading. *Node six* needs to have completed vessel grading sheets and a completed vessel log to perform the review. *Node seven* requires the output from node six to make a decision on which vessels to board. Finally, *node eight* requires that the boarding decisions be made before assigning a team to a vessel to be boarded. At any node, if the information is not available from the previous one, the activity of that node cannot be completed. It is possible to process each vessel arrival in parallel up to the point of node seven (making the decision on which vessels to board). This could be done by completely automating the activities from nodes one through six and allowing identical multiple processes to run in parallel up to the collection point of node seven. The bottleneck in the process would then be node seven, as it would need all of the inputs from the parallel processes to perform the boarding decision task.

Next is the case manager. A case manager is defined as a person who performs the majority of the activities in the process. The case manager would act as a single point of contact for the process and perform the process from beginning to the end. This eliminates the fragmentation of the process and provides continuity of thought and action through out the process. Implementing a case manager without performing any other intervention would likely be a mistake. If just a case manager were implemented, the majority of the process would fall on that one person's shoulders. The case manager could potentially perform all of the activities in the process except for node one, calling in a vessel arrival.

Having attempted to implement just such a system myself, it quickly became apparent that the reliance on one person's expertise to perform the entire process was unacceptable. The person performing the case manager role builds up a local knowledge of the process, has learned all of the rules for the process and knows what shortcuts are allowable and acceptable in the system. Additionally, when the entire process is performed by one person, the other personnel in the office look at that one person as being the "expert" and subsequently refer all questions in that subject matter to the "expert". Due to other operational programs, training, etc. they ignore the workings of the process and consequently do not have a working knowledge of the process when there is a designated expert. This leads to a rush to the manuals to learn the workings of the process when the "expert" is not present and errors are made that the "expert" would not have made. To continue my story, losing that one person led to confusion and multiple errors that then had to be sorted out at a higher level increasing the amount of time it took to dispatch boarding teams. Not only was there lost time, but the unit missed boarding a few high priority vessels due to the confusion and time it took to sort things out. Missing high priority vessel boardings is not something that was smiled upon by the Commanding Officer or his superiors. It was found through this experiment that having the ability to break down the responsibilities led to a more robust process when faced with multiple personnel absences, which were quite frequent due to the pull of other operational commitments. Therefore, a case manager could be implemented only if the majority of the process is automated, as the reliance on one person is unacceptable as discussed above.

Information technology is an area where the current process is particularly lacking. As seen with the IT related diagnoses from KOPeR, and a general perusal of the process description, the only IT related support provided to the process is a database system. While MSIS was state of the art at one point in time, it is currently unable to provide the type of service (e.g., 100% uptime) required by the PSC process and is at times highly unreliable. As an example, one month due to equipment problems, MSIS was not available for almost an entire week. Without MSIS, the PSC process quickly became a guessing game on which vessels to board as the vessel histories were not available.

Regarding IT support, as stated above, MSIS is the only IT support provided to the current process. However, the replacement for MSIS, the Marine Safety Network (MSN), is currently being developed. MSN is a relational database/application that will keep track of all vessel-related processes in the Coast Guard. It is designed to be accessed via the Internet from multiple locations. MSN can provide IT support to the process by supporting all of the process activities via use of an electronic repository of information which is easily accessible from any computer terminal in the Coast Guard using a web type interface. Other areas where MSN could provide IT support in conjunction with human labor are the grading of vessels (activity node four Figure 3-1) and the decision making process of deciding on which vessels to board (activity node seven Figure 3-1).

IT support and IT communication are closely related, though separate. In order to provide IT communications, a support function is required to hold the information that the communications function is providing. IT communication is also completely lacking in the current process. IT communication can be provided to the process with MSN just as IT support could be provided. IT communication would take the form of automating the keeping of lists of vessels (e.g., vessel arrivals, vessels targeted for boarding etc.) which could then be viewed by personnel associated with the process. This would eliminate the paper logs, and the passing of paper grading sheets and vessel histories.

With IT automation, there is a caveat that substantial infrastructure in terms of support and communication need to be made before automation can be put in place. As mentioned above, adding functionality to MSN can provide the required support and communication needed to provide additional automation to the process.

IT automation is concerned with removing the need for human labor in the process and having a computer perform the activities in the process. Extending the use of MSN a little further, the entire process, from the agent call in (activity node one) to the selection of vessels to board (activity node seven), could be automated. There would still be some related support and communication attributes to such a system, but the removal of human labor in most of the activities of the process would obviate the need for many of those attributes. Some specific ideas where IT automation could be applied are:

1. An application or module to MSN could be built that automatically scores vessels when the arrival is entered into the system.

2. A decision support/expert system consisting of rules developed from information mined from the database could be developed that assists in determining which vessels to board.

Additionally, utilizing IT automation would fill the gap needed to successfully implement the case manager as well as parallel processes as discussed above. IT automation is an area where I see the most pay off in the redesign of this process.

In addition to MSN, the Coast Guard has recently completed roll out of Coast Guard Standard Workstation III (CGSWIII). CGSWIII is a commercial off the shelf technology solution, which mirrors the IT-21 requirements of the Navy. It consists of a networked Windows NT operating system, operating on standard PC hardware. Each workstation has a standard system image (software and configuration), and has the ability to connect to the Coast Guard Intranet (CGWEB) as well as the Internet.

Leveraging the investment of CGWSIII and MSN to provide all IT attributes to the PSC process, as discussed above, appears to be the best way to proceed. There are some limiting factors to this such as no choice of hardware architecture, and the ways technologies could be used. For instance, since there is already a centralized database (MSN) in place, the use of client/server architecture has already been predetermined. Regarding the limitations of technology, the use of a local workflow system would not make sense due to the centralized database and client/server architecture.

To summarize this section, I will present a look at the redesigned process in diagnosis form. To delinearize the process, the activities of the process from the agent call in to the selection of vessels to board (activity nodes one through six) have been automated, which allows several of these processes to operate in parallel.

A case manager is implemented reducing hand off friction. This is due to the automation of the call in and grading process. The duties of the case manager now consist of selecting the vessels to board (with assistance from the decision support system) and assigning boarding teams to vessels. IT support is provided in the form of MSN keeping the information regarding vessel arrivals and vessel grading information. IT communication is provided by MSN through the CGWEB giving access to the vessel lists. Finally, IT automation is provided by automating the process and providing decision support to the case manager.

3. Benefits from Improving the Process

There are some benefits from improving this process. Only the measurable ones will be addressed here, as some of the other benefits obtained by improving the process may not be discovered due to their intangibility. For example, an improvement in morale that leads to better recruiting for the Coast Guard. Most of the benefits derive from the use of IT automation; however, there are some benefits deriving from the other diagnosis categories, but there are not as many due to IT automation being the primary enabler of this process redesign. I will begin with those categories that have the least number of benefits and finish with IT automation which has the most benefits.

A benefit gained from delinearizing the process would be a reduction in cycle time due to the addition of the automation, and the ability to process multiple vessel arrivals at one time.

A benefit gained from implementing a case manager would be the reduction of personnel from the process. Implementing a case manager would remove the personnel required for activity nodes two through five in Figure 3-1. These savings would be two people, the watchstander and the petty officer.

Replacing MSIS with MSN and adding additional IT support from MSN to the process would yield increased benefits by leveraging the investments of MSN and CGSWIII to a greater degree. Another benefit of IT support is that it is a key enabler of implementing the automation of the process. Without the IT support provided by MSN, the automation of the process would be more expensive to implement because a support mechanism (a distributed database) would have to be provided in addition to the automation itself.

A benefit of IT communication is that it is the second enabler to IT automation. IT communication allows automation to work more effectively because it eliminates the passing of paper. For example, eliminating vessel logs and grading sheets and letting automation electronically access all of the information contained in the logs. Another benefit is that vessel information is readily available to everyone needing it as opposed to one centralized place where only a few have access to it.

The benefits of IT automation are:

1. Automating a large portion of the process, if not all of it, results in removing an administrative burden from at least two personnel in the process, the supervisor and the person assigned to perform the vessel grading. This would also allow the implementation of a case manager.
2. Automating the grading of the vessels would remove this repetitive activity from human hands. The grading activity occurs for every vessel arrival at a port, for example, if there are 20 vessel arrivals a day one person has to print out histories and grade 20 vessels. Performing this task, day in and day out, 365 days a year, is a repetitive task best left to a computer. Assuming the automated grading process was implemented correctly, the supervisor would not have to review the grading results as closely as they would be correct every time. This would free the supervisor and person assigned to do the grading to perform other duties not related to the process. Therefore, the use of automation for this process would reduce rework and review time for the supervisor, as the information provided by the automated system would be more accurate and consistent.
3. Consistency in the application of the process throughout the Coast Guard would also improve with the automation of the process, as I will describe. With all units in the Coast Guard using the same system to process vessel arrivals from the beginning to the end of the process, there would be no ambiguity in the application of the business rules. Each vessel targeting process at each unit would be performed consistently as they would all be running off the same server running the same software. Therefore, a vessel that scores a priority one in Seattle would score as a priority one in Los Angeles as well. This is different from the current system as some MSOs score vessels differently.

4. Feedback to vessel agents and the vessels they represent would improve because when the targeting process is automated, the grading process would already be complete when the personnel come to work in the morning. The system will already have a list of suggested vessels to board for the supervisor, and the vessel agent can be notified his vessel is going to be boarded before he has his coffee in the morning. With this improvement in the availability of information, there will be a reduction in the process cycle time as shown above.

C. VESSEL BOARDING PROCESS MODEL

This section discusses the vessel boarding process model. The process model is described, with an eye toward identifying activity inputs and outputs, and depicted using a graphical method. Improvements and benefits to the process are then discussed later in this section. Figure 3-2 is a graphical depiction of the current targeting process. As I did in Section B, I will use the activities presented in Figure 3-2 to guide my discussion of the current vessel boarding process. Each activity is identified by a node number and is connected to the next node with a directed edge. In my discussion, I will identify the node number as well as the activity name to provide clarity in the discussion of the activity.

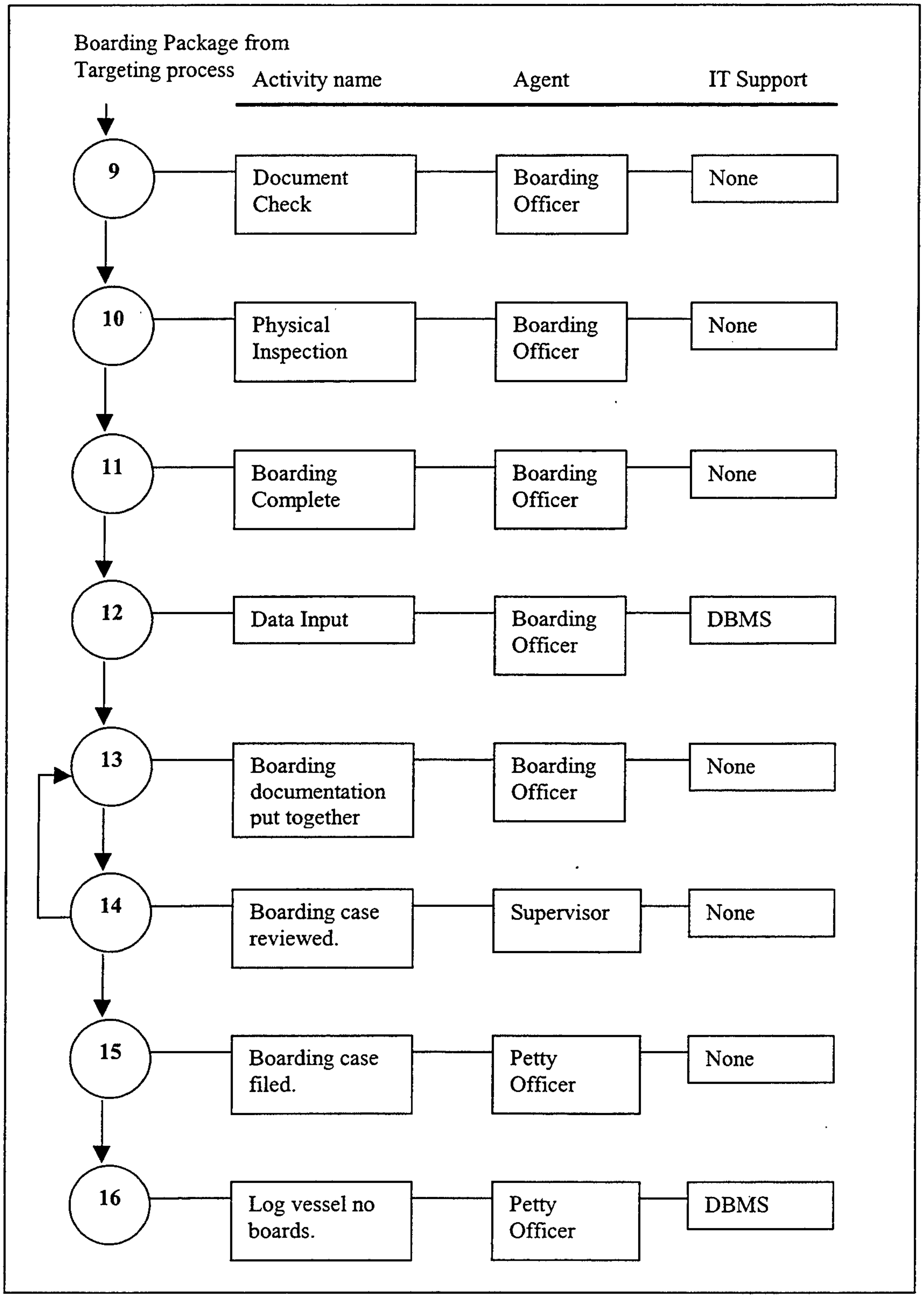


Figure 3-2. PSC Vessel Boarding Process

1. Current Process Model

This process starts at activity *node nine*, document check. The boarding teams are given the histories and completed grading sheets for the vessels assigned to the team. On board the vessel, the team uses a paper based boarding book to document the vessel boarding. Information regarding the vessel such as name, official number, date, and other information are hand copied from the vessel history to the boarding book. The boarding officer reviews the vessel's papers and compares the information contained on them to the vessel history. These papers are documentation, from the Flag State of the vessel, that the vessel has undergone the required safety and structural surveys called for by International treaties and laws as well as other physical description papers of the vessel. The primary information provided by these documents are issue dates, expiration dates, endorsement dates, and the entity issuing the document. Any changes to the information are recorded in the boarding book for the vessel. The vessel paperwork review is usually accomplished prior to any physical inspection of the vessel; this is to ensure the vessel actually needs a boarding and to give the vessel Master time to get the crew members ready to assist the boarding team.

Activity *node ten*, physical inspection. After the paperwork review, the physical inspection of the vessel is conducted, and the various inspection actions are initialed by the boarding officer signifying completion of the item(s).

Activity *node 11*, boarding complete. At the end of the physical inspection, any discrepancies found are transcribed from the boarding book onto a vessel boarding letter. The vessel boarding letter contains, in addition to any discrepancies, the identifying information on the vessel, the date of the boarding, and the signatures of the boarding officer and vessel Master. This letter is left on board the vessel as an official record of the results of the boarding. If there are any discrepancies such that the vessel would endanger the personnel on board or the environment, the vessel is held in port until the items are repaired. If this is the case, several other paper documents have to be prepared to hold the vessel and notify the chain of command about the detention. This paper work is usually done after the boarding team returns to the office. The letter(s) requires the signature of the Commanding Officer of the MSO and an additional trip to the vessel to deliver the official letter, which holds the vessel in port.

Activity *node 12*, data input. After leaving the vessel, the boarding officer prepares the paperwork documenting the boarding. The boarding book is checked for completeness, which consists of ensuring all items are initialed, all blanks are filled out, and all personnel in the boarding party have signed the boarding book. MSIS is then updated through the case number for the boarding. Any changes to vessel information are updated and a narrative of what was done on the vessel is completed. The updated information is printed out in hard copy form for inclusion with the boarding documentation.

Activity *node 13*, boarding documentation put together. The print outs, the boarding book, a copy of the boarding letter, and the detention paperwork, if the vessel was detained, is then compiled and submitted for review by a supervisor.

Activity *node 14*, boarding case reviewed. The supervisor reviews the boarding case and if any discrepancies in the paperwork are identified, they are noted and returned to the boarding officer for rework.

Activity *node 15*, boarding case filed. After final approval of the package, it is filed locally at the MSO. Then the case in MSIS is "validated", meaning that it is closed and further alterations to the record are not allowed.

Activity *node 16*, log vessel no boards. Finishing the process, vessels not targeted for boarding are tracked on when they leave the port. After a vessel departs, an entry is made in MSIS via the case number for the port call of the vessel. This closes the case in MSIS and provides documentation that the vessel made a port call, but was not boarded.

The depiction of the PSC vessel boarding process in Figure 3-2, in addition to providing a template for discussion, is also suitable for taking measurements for input into KOPeR. The attributes and other features are the same as presented in Section B. The measurements for the process that KOPeR needs in order to complete a redesign recommendation are shown in Table 3-4. The definitions and implications of the measurements are the same as those described previously for the targeting process.

Configuration Measure	Value
Process Size	8
Process Length	8
Handoffs	3
Feedback loops	1
IT-Support	2
IT-Communication	0
IT-Automation	0

Table 3-4. Measurements for the PSC Vessel Boarding Process

2. Possible Ways to Improve the Process

Giving the process measurements in Table 3-4 to KOPeR results in a diagnosis and a list of recommendations. As seen with the targeting process, the diagnosis provides the pathologies that were detected from the measurements. The diagnostic output from KOPeR is provided in Table 3-5.

KOPeR Diagnosis for the Vessel Boarding Process
Measurements (e.g., size of 8) suggest the *small Vessel Boarding Process* suffers from the following pathologies: • Parallelism (1.0) - *sequential process.* • Handoffs fraction (0.375) - *process friction.* • Feedback fraction (0.125) - *feedback looks OK.* • IT support fraction (0.25) - *inadequate IT support.* • IT communication fraction (0.0) - *inadequate IT communications.* • IT automation fraction (0.0) - *IT automation first requires substantial infrastructure in terms of support and communication.*

Table 3-5. KOPeR Diagnosis (From KOPeR Web Page)

Inspection of the diagnosis shows the pathologies identified are very similar to those found for the targeting process. The measures, how the numbers are calculated, and the performance implications are the same as those discussed previously in Section B for the targeting process.

Again, as seen in Section B, KOPeR then provides a set of recommendations for transformations to the process based on the pathologies identified above. These recommendations are contained in Table 3-6.

KOPeR Recommendations for the Vessel Boarding Process
Delinearize process activities to increase parallelism; such activities must be sequentially independent (e.g., have mutually-exclusive inputs and outputs). *Try a case manager or **case team** to decrease friction; be sure to include a source of expertise.* *Look to information technology to increase **support** to process activities; decision support systems and desktop office tools generally have good payoffs and intelligent systems can greatly enhance knowledge work; be sure to address personnel training and maintenance of the IT.* *Look to information technology to increase support to process **communications**; e-mail and shared databases through local/wide area networks generally have good payoffs and workflow systems can greatly expedite process flows; be sure to address personnel training and maintenance of the IT.* *Look to information technology to **automate** process activities, but note that substantial IT infrastructure is first required, particularly in terms of process support and communication; try workflow systems for support and communication, and then look to intelligent agents, which can enable many electronic commerce opportunities.* *In addition to delinearization and the use of a case manager, workflow systems offer good potential for process improvement; this requires substantial IT infrastructure and support however.*

Table 3-6. KOPeR Recommendations (From KOPeR Web Page)

As stated in Section B, the results from KOPeR are generic and need to be specific to complete the redesign. The recommendations are markedly similar to those of the targeting process; this is not surprising as the pathologies found for both processes were identical. Examination of the process in light of each recommendation will bring out some ways of redesigning the process to improve performance.

Proceeding in the same fashion as in Section B, the inputs and outputs of each activity node will be examined to determine if the activities are sequentially independent. Node numbers are taken from Figure 3-2. The process begins with *node ni*ne--the document check. Regarding its input, there is the vessel history, and the output there is the boarding book entries resulting from the document check. *Node ten*--the physical inspection of the vessel. Here the boarding book is used, but is not necessary for the activity to begin. *Node 11* requires the inputs from nodes nine and ten to complete the boarding. *Node 12* requires the output from node 11 and has as its output the raw materials for the next activity at node 13. *Node 13* requires the raw materials from node 12 to complete the boarding package and pass it on to the supervisor for review in node 14. The supervisor needs to have a package to review at *node 14*, and *node 15* needs the approved package to complete the filing process. Finally, *node 16*--logging vessel no boards, does not rely on any of the nodes in this process and arguably could be a separate process in itself. The input to node 16 is a vessel departure notice. This activity was included in this particular process as the practice in the field is to update MSIS items in batches due to the slow speed and the availability of computer terminals. This task, referred to as "feeding the green eyed monster" (due to the monochrome green monitor), is particularly dreaded by marine safety personnel due to the monotonous nature of the task. When the finished boarding cases are validated in MSIS, the vessels not boarded, which have departed the port, are updated at the same time.

For the process, the only two nodes that could run in parallel are nodes nine and ten. This is because it is not necessary to complete the document check before beginning the physical inspection of the vessel. However, some additional background on the process is necessary to understand why the document check (node nine) and the physical inspection (node ten) are sequentially positioned. In the International treaties, a portion states that parties to the convention will accept the attestation of the Flag State that the vessel complies. Due to the shabby state of some vessels entering US waters, the Coast

Guard extended this to include a physical examination to verify the attestation of the Flag State. To give the boarding officers a better feel of some of the areas to concentrate on during the physical inspection, the document examination is done first. The document check activity also gives the boarding officer a feel for how well the crew interacts with each other and provides time to explain to the Master the procedures of the examination. The time saved in making the activities parallel would not offset the information gained by leaving the activities serial. For these reasons, I do not recommend changing the flow of the process to have these two activities run in parallel.

While KOPeR, in its diagnosis, states the process has friction in the forms of handoffs, inspection of the process shows the boarding officer does the majority of the work. In essence, the boarding officer is acting as a case manager. The boarding officer performs the majority of the process and only hands off the final documentation to the supervisor for review. The review is an important item in the process as it provides a final check on the completeness of the work done. Keeping the official record of the boarding electronically could easily eliminate the hand off between the supervisor and a filing clerk. This would require a technology solution to implement and may require portions of the process to be automated; further discussion of automation will follow. The final hand off seen between the filing activity (node 15) and the log vessel no boards (node 16), should not be counted in the analysis of this process. As discussed above, the log vessel no boards activity is a separate process; therefore, the friction from this hand off should not count in the analysis of the vessel boarding process.

Again, the remaining recommendations involve information technology. IT support for the vessel boarding process is, again, supplied by MSIS. This support is only supplied while the boarding team is in the office; there is no IT support while the boarding team is on the vessel. MSN could provide the same type of support that MSIS provides currently; however, IT support should be provided for the entire process. IT support for the remote portions of the vessel boarding process could be provided in a couple different ways. Providing the IT support would require either a connection to the

Internet/CGWEB via a portable computer or some sort of portable database that could be synchronized when the boarding team returns to the office.

IT communication could be added to the process by leveraging MSN as was proposed in Section B, especially for those parts of the process that take place in the office. Linking the boarding team on a vessel and the main office would also provide IT communication as the linking would help to eliminate paper passing in the process. Providing a link between the boarding officer on a vessel and the main office is an area where the current IT infrastructure of the Coast Guard is lacking. This is due to the complete absence of any technology to provide a link other than the cellular telephone or radio. There are two ways, wireless and a portable application, to provide the IT communication connection between the office and boarding team.

First, wireless connections to the Internet are currently the "rage" in technology publications. These wireless connections can take two different routes to the connection. The first route is via a cellular telephone. There are limitations to this avenue, the first being the data rate achievable over such a connection, and the other being dead spots in the coverage areas. As anyone who has used a cellular telephone can attest, "dead spots" are quite common the farther away one moves from urban environments. The dead spots are the downside for the cellular telephone. Although the coverage for cellular telephones is usually very good in metropolitan areas, there are numerous areas (Alaska, the southern coast of Oregon and distant offshore anchorages) where the Coast Guard performs boardings that are far from these metropolitan areas.

The second route would be a wireless option using a technology compliant with the IEEE 802.11 standard or some other proprietary wireless solution. The advertised range of these solutions is from 600 to 1200 feet in open areas. (Orinoco FAQ web page) Most of the proven implementations of these technologies have focused on limited geographic areas such as a college campus, warehouse or hospital complex. (BreezeCom Solutions web page) As stated for the cellular telephone, the locations of vessel boardings are very diverse. While the vendors of these products are constantly researching ways to enhance the range of the technology, using "bleeding edge" technology for a production system is not cost effective due to the lack of maturity of the product, as well as the costs involved with using very advanced technology (look at the cost of computer systems with the latest processor compared to those with processors one or two generations older). Additionally, due to the cost (hardware, maintenance and spares), setting up a wireless solution in multiple port areas would be expensive and would not cover all areas where vessel boardings are conducted. While promising, it would be wise to let the technology mature in this area.

The second approach would be a portable computer with a custom software application that would emulate the MSN database and would be able to synchronize the data upon return to the office. This would provide IT communication, as well as IT support, while on board the vessel. The documentation could be kept online, and boarding officers could have palm size devices to use as electronic note pads. They could write comments and then via infrared connections add the comments to the main documentation. Making all documentation electronic would allow the reduction of some hand off friction in the process. While still not a low cost option, the costs involved in developing and fielding a system described above would still be less than those required to establish true wireless connection in all of the areas the Coast Guard performs boardings. Based on the lower costs of this type of implementation, I recommend this option.

IT automation could be provided to the process by electronic filing of boarding records and by providing a portable printer and additional functionality to the portable application. Required paperwork, such as the boarding letter, could be printed out instead of hand written. Providing automation to this process does not have as big a pay off as with the targeting process, but still provides some benefits of reducing cycle time.

As in Section B, a view of the redesigned process in diagnosis form is provided to gain insight on how a possible redesign addresses the diagnosis. This begins with parallelism. The process remains sequential; this is a requirement, as the extra information gained by the boarding officer is not offset by the time saved by running the activities on the vessel in parallel.

Next in the diagnosis is process friction. The boarding officer acts as a case manager already; he handles all of the activities in the process up to the review of his work. Removal of one handoff is accomplished by automating the filing activity at node 15 in Figure 3-2.

IT support, the next item in the diagnosis, is provided by MSN at the office and by a portable computer while on board a vessel. IT support consists of keeping track of vessel information and automated note keeping while on board the vessel.

Following IT support is IT communication which is provided by MSN and the portable computer. Vessel boarding cases are created online and passed to the supervisor via the MSN.

The final diagnosis is IT automation. This is provided by the portable computer generating the boarding letters and by completing the boarding documentation by passing it to the supervisor.

3. Benefits from Improving the Process

There are several benefits to be gained by improving the process. Only those benefits that are readily apparent and tangible will be addressed in the section. Other benefits that are less tangible may be discovered after the implementation of the redesign, as addressed previously in Section B.3. Most of the benefits in this process derive from the use of IT support and IT communication. In this case there are fewer benefits to be gained from IT automation. Since the process is not delinearized there are no benefits to be gained. Regarding the case manager, the process already incorporates one as discussed above in Section C.2. The discussion begins with benefits of IT support followed by IT communication and IT automation.

Regarding the benefits of IT support, the first deals with consistency. Having boarding officers use the same method of documenting a boarding will allow the Coast Guard to improve the consistency of application of the entire boarding process. This is due to the fact that every boarding team would be using the same software and hardware packages to perform the process. Another benefit is the increased availability of information. Near instantaneous data input to the database system will allow other MSOs to target the vessels on the most up to date data, reducing the number of unneeded visits to vessels.

Regarding the benefits of IT communication, the first deals with data integrity. Reducing the number of times a boarding officer writes the same information will increase the accuracy of the data gathered, as well as save time during the boarding of the vessel. As seen with the copying of manuscripts before the advent of the printing press, several different types of errors could be made. From outright errors in copying the information, to less visible errors such as changing the meaning of what is being copied by slightly changing the wording; these errors are usually present in any manual form of copying information. For example the translation of the Bible from the original Hebrew to the King James Version. Capturing the data once will help to eliminate these errors, thus reducing the total number of errors in the process. Additionally, the time saved by entering the data once will cut down on the time spent on the vessel performing administrative tasks, such as copying notes to the boarding book, and preparing the vessel boarding letter.

Another benefit of IT communication is reduction in cycle time. Documenting the boarding online on a portable computer while still on board the vessel will speed the review of the boarding documentation. This will eliminate the lag between the boarding officer returning to the office, and the boarding paperwork being put together and passed on to the supervisor. In the current process, this lag time can be as long as three days. Having the boarding documentation complete upon return will allow the entry of the information to proceed immediately without having to wait for the boarding officer to compile and turn in the case work.

Finally, the benefits of IT automation are reduction in cycle time and reduction of handoff friction. The reduction in cycle time stems from the use of a portable computer and printer to create the boarding letter left on the vessel. The electronic filing of the boarding documentation in the central database provides the reduction of handoff friction.

D. SUMMARY

This chapter has covered the current PSC process in two parts, the targeting process and the vessel boarding process. Each of the processes was described and then analyzed using measurement based inference to diagnose the BPR pathologies. From those pathologies, generic recommendations were provided which were then made more specific through the use of manual analysis from my first hand knowledge of the process. Finally, benefits of improving the processes were identified.

The most important part of this chapter, to carry on to the next chapter, is the recommendations on how to improve the processes. The recommendations are: use of client/server architecture, automation of the process, leveraging off current IT infrastructure, and a web enabled database. These are foundational for the redesign efforts of the processes. The next chapter presents the redesigned processes, and using simulation, provide some tangible evidence that the redesign will significantly improve the process.

IV. PROCESS REDESIGN

The two major PSC subprocesses have been redesigned based upon the diagnosis of the pathologies of the overall process presented in Chapter III. The redesign takes into account the fiscal constraints of the Coast Guard and the recommendations found in Chapter III. An explanation of the redesigned processes and presentation of simulation models for the current and redesigned processes are presented below.

A. REDESIGN OF THE TARGETING PROCESS

1. Model and Description of the Proposed Process

As in Chapter III for the current processes, I use the model of the proposed process to guide the discussion of the process activities. The proposed process model is presented as Figure 4-1.

Activity node 1: Enter vessel arrival information. The redesigned targeting process begins with the vessel agent entering the arrival of a vessel on a web page linked to the MSN database system. The information on the agent, the vessel, and the arrival date are captured and entered into the database.

Activity node 2: Vessel grading. At the server, the database is queried for additional vessel information, and the results of the query are submitted to a grading algorithm, which generates a grading profile to the agent via a dynamic web page. Based on the profile, the agent gains insight about whether the vessel is a likely candidate for boarding during the port call.

Activity node 3: Log grading results. The grading profile data are stored with the arrival information for later retrieval.

Activity node 4: Boarding decisions made. The supervisor of the PSC branch at the MSO accesses a web page that shows the vessels due to arrive and vessels already in port for their area of responsibility. The vessel information is listed by priority with the highest priority vessels listed first. Vessels with the same priorities are ranked by a decision support/expert system that uses data mining and/or a multi-criteria decision model to rank relevant risk items on the vessel and provide a recommendation on which vessels pose the most risk. The vessels selected for boardings that day are checked off, and the information is submitted to the server. The server returns a confirmation of the vessels selected and stores the information for further retrieval as well as for subsequent use in the boarding process.

Activity node 5: Boarding teams dispatched. Boarding teams are assigned to the vessels selected for boardings.

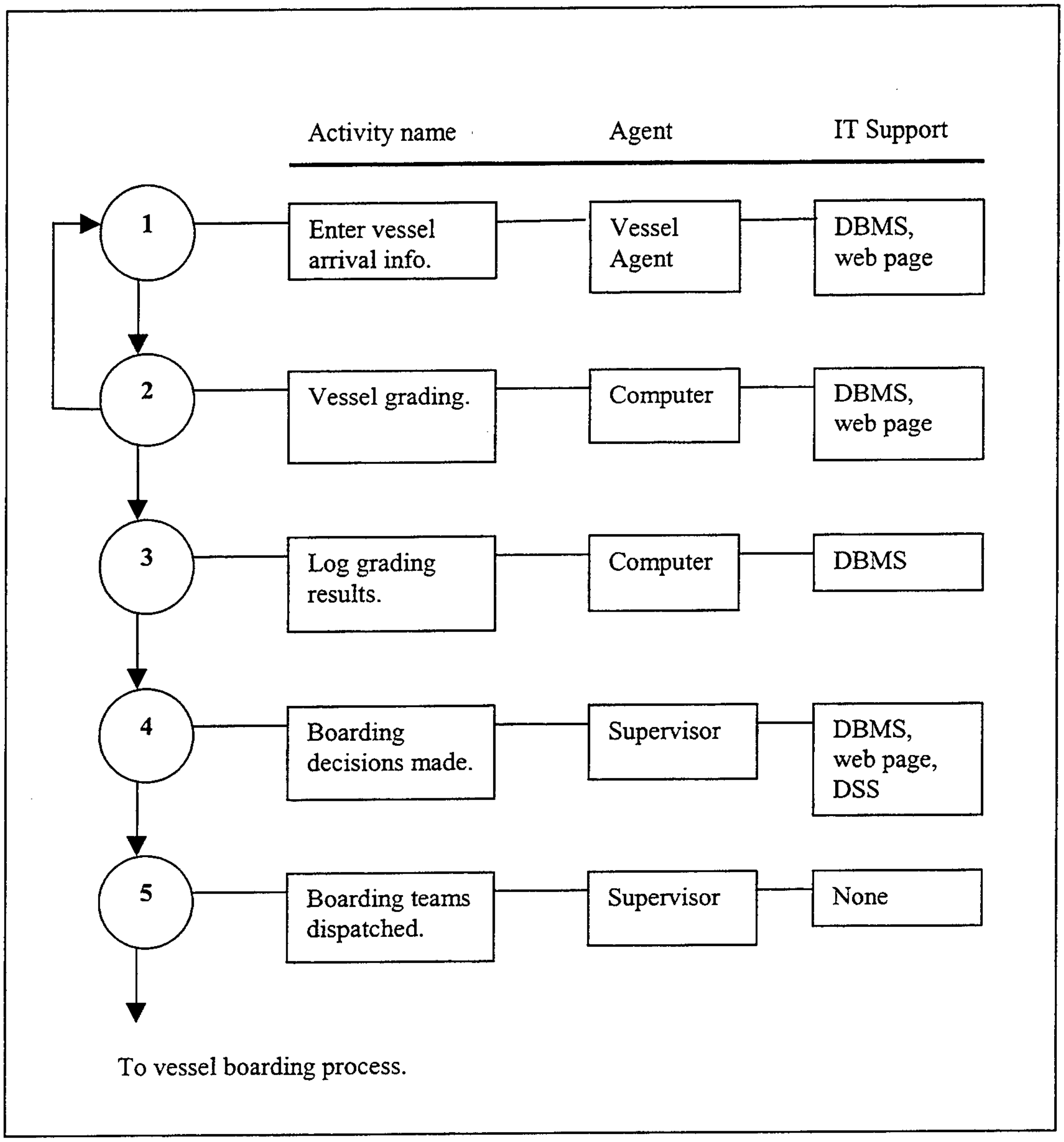


Figure 4-1. Redesigned Targeting Process

This redesign relies heavily upon information technology for successful implementation. The client/server architecture allows for inexpensive implementation, centralized maintenance, and facilitation of the familiar web browser interface to complete the process. Additionally, the use of the client/server architecture leverages the investment the Coast Guard has made in the Standard Workstation III contract.

The redesign described above is, in my opinion, an optimal redesign for the process, and is the one modeled in this Chapter. However, I have provided two variations of the process, which incorporate the core elements of the redesign because I see two areas of resistance by the Coast Guard and industry regarding the full implementation of the redesigned process.

The first area of resistance stems from the vessel agents entering the vessel arrivals on a web page as opposed to using a telephone to call in the arrival. To overcome this resistance, a possible variation on the redesigned process is to eliminate the requirement of entering the vessel arrival information on a web page. This variation would have the agent call in the arrival information to a watchstander, as in the current process, who then uses a web page to enter the data in the database for grading. In this case, the beginning of the process would be the same as the current process so the vessel agent sees no change.

The second area of resistance may be from those ports that have a centralized broker. These brokers take the arrival information from the vessel agents and then provide the information to the Coast Guard under a special agreement. In most cases, the centralized brokers have a more involved relationship with the Coast Guard than just providing vessel information such as providing vessel traffic control. Removing this one service may damage the relationship between the Coast Guard and the broker. In this case, the variation would be that vessel arrivals would still be provided by the broker, thus reinforcing the relationship. The vessel arrivals would be entered into the database via the web page by a petty officer in the PSC branch.

To help quantify the advantages of the redesign, simulation models of the current and proposed processes have been created in EXTEND+BPR®. These models are designed to capture the critical performance measures of the process in a consistent manner to allow meaningful comparisons of the two processes. Unless noted otherwise, the parameters used in the models are estimated from three years (1996-1998) of my personal experience (i.e., as a subject matter expert) in directly performing and supervising the PSC process. This is a limitation that is discussed later in this Chapter.

Figures 4-2 and 4-3 present portions of an EXTEND+BPR® simulation model covering the targeting process. Several major assumptions have been made in the design of the model, which will be identified during the course of the step-by-step model discussion.

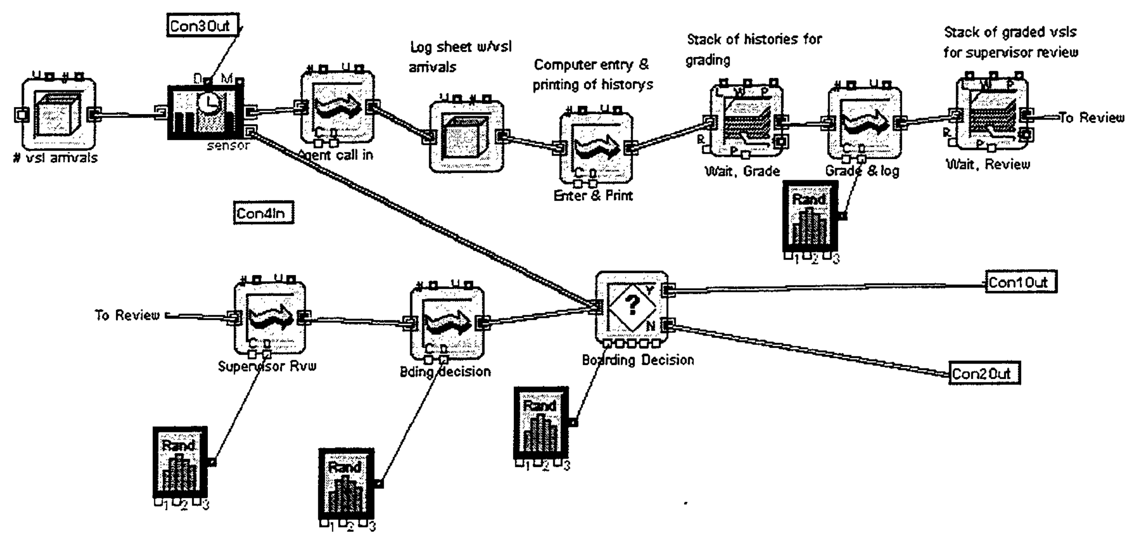


Figure 4-2. Current PSC Targeting Simulation Model, Part 1

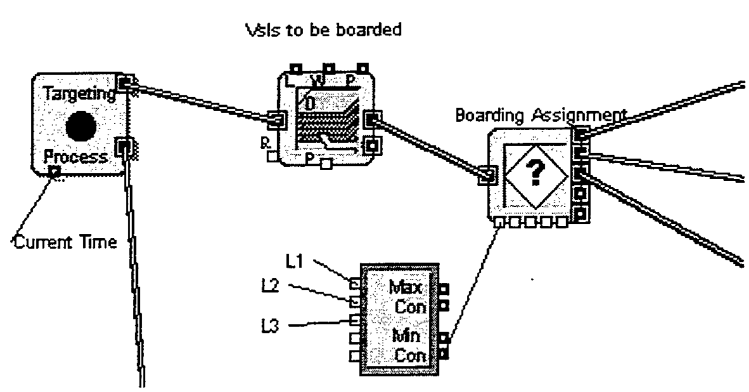


Figure 4-3. Current PSC Targeting Simulation Model, Part 2

The model starts in step one with a repository of vessels randomly selected from a uniform distribution with a minimum and maximum of two and 15, respectively. This simulates an average daily number of arrivals for a medium to large sized port such as Los Angeles or New Orleans. Ports of this size will likely benefit the most from the redesign due to the larger number of vessel arrivals, and subsequently the greater amount of time spent performing the PSC process.

Step two consists of a timer, which allows for the computation of cycle time for each vessel moving through the system.

Step three is a transaction block, which represents the agent calling in the vessel arrival to a watchstander. The time for this transaction is assumed to be fixed at three minutes per vessel, due to the routine nature of the information passed and the familiarity of the vessel agents with the information requirements.

In step four the information flows to a holding or repository block, which represents the log sheet the watchstander has filled out containing the arrival information.

Step five is a transaction block that simulates the time it takes for the entry of information into the MSIS system. The time for this transaction is assumed to be fixed at three minutes per vessel, again because this is a routine activity with very structured information requirements.

Step six is the First In, First Out (FIFO) queue, which simulates the stack of vessel histories waiting to be graded, on a first item in first item out basis.

Step seven is a transaction block that simulates the grading of the vessels. The time for the grading of the vessel is randomly assigned from a real, uniform distribution with min and max values of two and five minutes per vessel, respectively. This assumption captures the varying nature of vessel histories, and the human factor involved in the grading of the vessels.

Step eight is the transaction block that simulates the supervisor's review of the vessel grading sheets. Again, the time for this block is assigned from a real, uniform distribution with min and max values of two and eight minutes per vessel, respectively.

This assumption simulates the range of mistakes that may be made filling out the grading sheets, and the associated complexity of the information being reviewed.

Step nine, the boarding decision, is broken into two different blocks, A and B, one for the time it takes to make the boarding decision, and one for implementing the decision. Block A is the transaction block that simulates the time it takes to make the boarding decision on each vessel. The time interval for the transaction block for the decision is assumed to be between one and five minutes per vessel from a real, uniform distribution. This takes into account the time it may take to decide between two or more vessels. Block B, the decision block, does not have a time delay. This block is set up to provide a boarding ratio of 0.25. This means that for ten vessel arrivals, one fourth of those vessels will be selected for a boarding. The decision block is set up to send a vessel to the boarding area of the simulation, if the random number provided as the input is 0.25 or less; otherwise, the vessel is sent to the not boarded area of the simulation. Since the decision to board is based on a random number from a real, uniform distribution, a simulation run may see a boarding ratio that is more or less than 0.25. This simulates the average USCG boarding rate of vessels calling at US ports for the year of 1998, the last year for which complete statistics are available at this time. Although the 0.25 ratio is the simulation average for USCG boardings, this should not imply there is always the manpower available to board all the vessels needing a boarding that arrive the same day. Not boarding a vessel the same day it arrives is acceptable as vessels usually stay in port for a period longer than one day. In addition, it should be noted that the boarding ratio of 0.25 is not a mandated target, but rather a naturally occurring phenomenon. The Coast Guard has a requirement to board vessels at a six month interval, and the 0.25 boarding ratio seems to occur naturally because of this, as the ratio consistently appears on annual reports, both nationally and at local unit levels.

Step ten is the final part of the simulation model. This decision block simulates the assignment of boarding teams. It is assumed that there is no time delay, as the tracking of personnel lies outside the scope of this thesis. A summary of the parameters for the blocks in the simulation model is presented in Table 4-1.

Block Name	Value/Range	Fixed/Random	Distribution
# Vessel Arrivals	(2.0, 15.0)	random	Integer, Uniform
Agent Call in	3	fixed	N/A
Enter & Print	3	fixed	N/A
Grade & Log	(2.0, 5.0)	random	Real, Uniform
Supervisor Rvw	(2.0, 8.0)	random	Real, Uniform
Bdng Decision	(1.0, 5.0)	random	Real, Uniform
Boarding Decision	(0.0, 1.0)	random	Real, Uniform

Table 4-1. Current Targeting Process Simulation Model Parameters

A final assumption, not explicit in the model itself (due to the fact that this variable(s) is extremely unpredictable), is that the personnel working on the process are focused solely on completing the tasks of the process with no interruptions. Modeling of this type of uncertainty is not needed for the comparison purposes of cycle time, but is necessary to mention, as it is a possible limitation of the simulation model. Additionally, since controlling interruptions to the personnel performing the process is very difficult in the real world, modeling the uncertainty of this variable would serve best as a topic of a separate thesis.

At this point the simulation model is used to compute the baseline average cycle time of the current process, one of the critical performance measures identified in Chapter II. Ten runs of the simulation model described above are conducted. This means that a random number of vessel arrivals will run through the model for each of the ten runs. This would simulate ten days in the life of the PSC process. The full data for the simulation runs as well as the statistics (mean, max, min and standard deviation) for each run, and the overall statistics are presented in Table 4-2. Each column in the table represents a run of the simulation model. Each cell in each column represents a vessel arrival, and the time it took for each vessel to move through the simulation model. Discussion and interpretation of the results are covered later in this chapter.

Sim #	1	2	3	4	5	6	7	8	9	10
Ship #1	15.19	15.81	20.97	19.38	10.33	16.50	16.54	16.72	17.13	14.81
Ship #2	13.56	19.78	17.01	16.72	12.14	18.83	18.23	17.27	15.27	15.63
Ship #3	14.39	19.12		9.98		15.88	20.35	19.11	17.79	13.81
Ship #4	20.84			17.24		13.39		12.77	17.45	15.73
Ship #5	14.86			15.62				15.79	16.14	16.15
Ship #6	19.89			16.95				15.47	19.58	17.18
Ship #7	14.07			13.89				15.43	16.22	16.72
Ship #8	14.78			17.12				13.39	19.93	16.47
Ship #9	13.22			16.85				15.41		15.07
Ship #10	12.39			15.12						11.65
Ship #11	17.68			18.68						16.22
Ship #12										14.40
Ship #13										17.65
Ship #14										19.59
Ship #15										18.71
Mean:	15.32	18.24	18.99	15.88	11.23	16.15	18.38	15.71	17.44	15.32
Max:	20.84	19.78	20.97	19.38	12.14	18.83	20.35	19.11	19.93	17.18
Min:	12.39	15.81	17.01	9.98	10.33	13.39	16.54	12.77	15.27	11.65
Std Dev:	2.80	2.13	2.80	2.53	1.28	2.24	1.91	1.91	1.64	1.63
Totals:	**Mean:**	16.08	**Max:**	20.97	**Min:**	9.98	**SDev:**	2.46		

Table 4-2. Simulation Delay Times for the Current Targeting Process

The other critical performance measure identified in Chapter II is data accuracy. Data accuracy is measured by counting the number of times data relating to a vessel is copied to another place, either on paper or as data input into a computer database. As borne out in experience with manuscripts before the advent of the printing press, it is assumed that the smaller the number of transcriptions, the more accurate the data. This is more of a qualitative measure as opposed to a quantitative one. The value of the data accuracy measure for the current targeting process is four.

Figure 4-4 is a portion of the EXTEND+BPR® simulation model for the redesigned targeting process. As above, the assumptions of the model are explained by following the flow of information through the model.

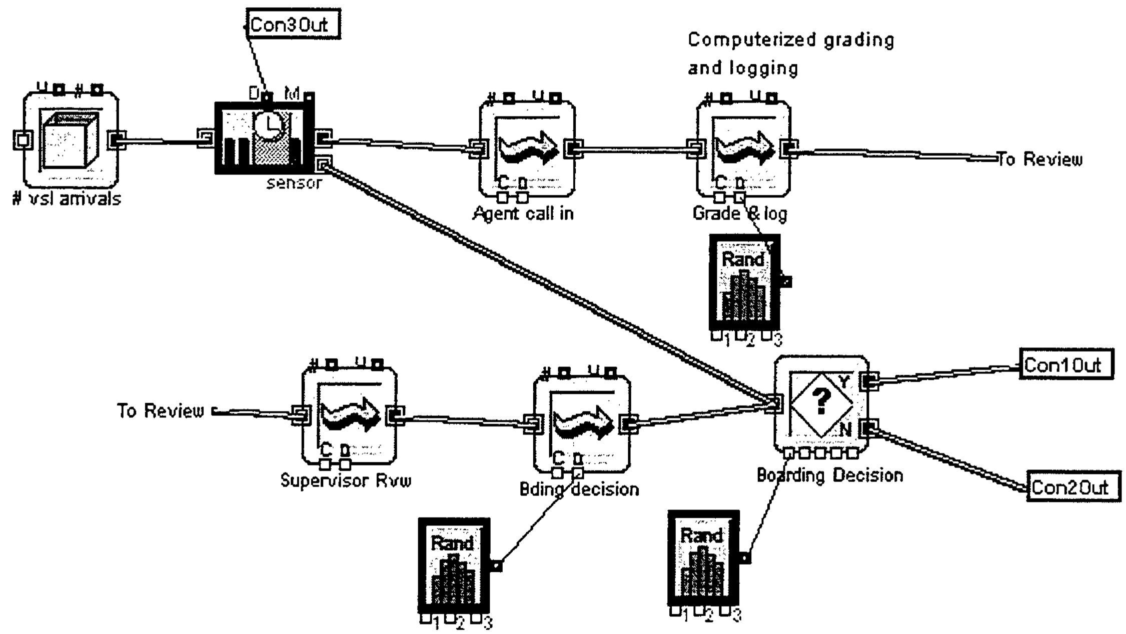


Figure 4-4. Simulation Model for the Redesigned Targeting Process

This model begins with a repository of vessels, the number of which is randomly assigned exactly as done with the model of the current process.

Step two is a timer block which allows for the computation of cycle time for each vessel moving through the system.

Step three is the first transaction block. This simulates the agent logging the arrival information into the web page. This activity is assumed to be a constant of three minutes per vessel to account for the agent logging in and then entering the pertinent information.

Step four is the second transaction block. This simulates the server running the grading and logging process. This process is assumed to require a time randomly selected from a uniform distribution with a minimum and maximum of one and two minutes, respectively. This is to account for varying traffic loads on the server, bandwidth and complexity of the vessel record.

Step five is the third transaction block. This block simulates the supervisor reviewing the vessels for boarding. Since this is a computerized list, the time to complete the review of the list is fixed at one minute per vessel.

Step six is the boarding decision. Like the current process model, this step is implemented with two blocks, A and B. Block A is the boarding decision transaction block, which is assumed to have a delay time of one to three minutes per vessel. This accounts for the time taken by a human supervisor to decide on the vessels to board even when given a list supported by a decision support system. Block B is the actual boarding decision block. It is configured the same as the boarding decision block for the current simulation model.

The rest of the simulation model is identical to the model for the current simulation model. A summary of the parameters for this simulation model is presented in Table 4-3.

Block Name	**Value/Range**	**Fixed/Random**	**Distribution**
# Vessel Arrivals	(2.0, 15.0)	random	Integer, Uniform
Agent Call in	3	fixed	N/A
Grade & Log	(1.0, 2.0)	random	Real, Uniform
Supervisor Rvw	1	fixed	N/A
Bdng Decision, time	(1.0, 3.0)	random	Real, Uniform
Boarding Decision	(0.0, 1.0)	random	Real, Uniform

Table 4-3. Redesign Targeting Process Simulation Model Parameters, Most Likely Scenario

At this point, the simulation model is used to compute the average cycle time of the redesigned process. To provide a full range of simulation numbers, three different scenarios were developed, *most likely*, *optimistic* and *pessimistic*. The parameters for the most likely scenario were determined by estimating a range of the time saved for each activity. Based off the most likely scenario parameters, high and low limits for each activity were estimated, thus providing the set of parameters for the optimistic and pessimistic scenarios. Again, these estimates were based on my own personal experience, stated earlier in this Chapter. The parameters for the most likely scenario are presented in Table 4-3. The parameters for the optimistic and pessimistic scenarios are provided as Tables 4-4 and 4-5 respectively. For each scenario, ten runs of the simulation were conducted. This was done to get the process cycle times for comparison with the baseline number. The full data for the most likely, optimistic and pessimistic scenario runs are presented in Tables 4-6, 4-7 and 4-8 respectively.

Block Name	Value/Range	Fixed/Random	Distribution
# Vessel Arrivals	(2.0, 15.0)	Random	Integer, Uniform
Agent Call in	2	Fixed	N/A
Grade & Log	(0.5, 1.5)	Random	Real, Uniform
Supervisor Rvw	0.5	Fixed	N/A
Bdng Decision, time	(0.5, 1.5)	Random	Real, Uniform
Boarding Decision	(0.0, 1.0)	Random	Real, Uniform

Table 4-4. Redesign Targeting Process Simulation Model Parameters, Optimistic Scenario

Block Name	Value/Range	Fixed/Random	Distribution
# Vessel Arrivals	(2.0, 15.0)	Random	Integer, Uniform
Agent Call in	4	Fixed	N/A
Grade & Log	(2.0, 3.0)	Random	Real, Uniform
Supervisor Rvw	2	Fixed	N/A
Bdng Decision, time	(2.0, 4.0)	Random	Real, Uniform
Boarding Decision	(0.0, 1.0)	Random	Real, Uniform

Table 4-5. Redesign Targeting Process Simulation Model Parameters, Pessimistic Scenario

Sim #	1	2	3	4	5	6	7	8	9	10
Ship #1	7.10	7.08	7.25	7.75	8.35	7.80	6.78	7.39	8.01	8.26
Ship #2	6.50	7.79	6.78	8.34	7.06	8.10	7.00	7.14	7.04	7.43
Ship #3		8.28	7.41	7.17	6.34	8.25	8.04	6.35	7.71	7.44
Ship #4		8.30	8.34	8.05	7.70	7.93	8.46	8.39	6.64	7.83
Ship #5		8.49	7.85	8.13	6.13		7.00	6.95	7.26	7.68
Ship #6		7.52	8.04	7.49	6.67		7.69	7.76	6.91	8.69
Ship #7		6.39	7.38	6.70	7.74		8.05	8.46	7.02	
Ship #8		7.89	8.08	6.53	7.72		7.90	8.29	7.87	
Ship #9		8.78		7.42			7.61	7.36	6.65	
Ship #10		7.59		8.22			7.82	8.47	7.15	
Ship #11		9.81						9.52		
Ship #12		7.36						9.34		
Ship #13		8.90								
Ship #14										
Ship #15										
Mean:	6.80	7.81	7.64	7.58	7.21	8.02	7.64	7.66	7.23	7.89
Max:	7.10	8.78	8.34	8.34	8.35	8.25	8.46	8.47	8.01	8.69
Min:	6.50	6.39	6.78	6.53	6.13	7.80	6.78	6.35	6.64	7.43
Std Dev:	0.42	0.71	0.52	0.63	0.79	0.20	0.54	0.74	0.49	0.50
Totals:	**Mean:**	7.58	**Max:**	8.78	**Min:**	6.13	**Sdev:**	0.64		

Table 4-6. Simulation Delay Times for the Redesigned Targeting Process, Most Likely Scenario

Sim #	1	2	3	4	5	6	7	8	9	10
Ship #1	4.06	5.00	4.24	4.00	4.07	3.75	4.08	4.05	4.61	4.49
Ship #2	4.85	3.98	5.01	4.85	4.47	4.70	4.54	5.13	5.32	4.50
Ship #3	4.07	4.72	4.79	4.33	4.60	5.31	4.83	4.31	4.20	4.35
Ship #4	5.21	4.15	5.18	3.97	5.16	4.26	4.19	4.68	4.46	4.85
Ship #5	4.54	4.52		4.33	5.19	4.97	4.83		5.02	4.14
Ship #6	4.55	4.43		5.00	4.98		4.57		3.94	4.72
Ship #7	3.84			4.34	4.77		4.73		4.58	4.39
Ship #8	4.26			4.32	3.92		3.69		3.98	
Ship #9	4.70			4.01	4.98		5.05		4.24	
Ship #10	3.92			4.60	4.18		4.05			
Ship #11	5.27						5.50			
Ship #12							5.51			
Ship #13							5.51			
Ship #14										
Ship #15										
Mean:	4.40	4.47	4.80	4.37	4.63	4.60	4.46	4.54	4.48	4.49
Max:	5.21	5.00	5.18	5.00	5.19	5.31	5.05	5.13	5.32	4.85
Min:	3.84	3.98	4.24	3.97	3.92	3.75	3.69	4.05	3.94	4.14
Std Dev:	0.44	0.37	0.41	0.35	0.46	0.61	0.43	0.47	0.46	0.24
Totals:	**Mean:**	4.50	**Max:**	5.32	**Min:**	3.69	**Sdev:**	0.42		

Table 4-7. Simulation Delay Times for the Redesigned Targeting Process, Optimistic Scenario

Sim #	1	2	3	4	5	6	7	8	9	10
Ship #1	12.12	11.84	11.94	10.90	11.12	12.08	11.29	12.50	10.90	12.14
Ship #2	12.15	11.16	11.93	10.64	12.67	10.86	12.62	12.16	12.10	11.37
Ship #3	11.88	11.55	11.80	10.83	11.53	12.07	11.77	11.88	10.82	10.32
Ship #4	10.81	11.17	11.50	10.89	12.91		10.66	11.78	11.50	11.52
Ship #5	10.72	10.66		10.28	10.65		11.47	11.87	11.11	11.86
Ship #6	11.57	12.09		11.51	12.47		12.09	11.78	11.75	11.10
Ship #7	11.47	10.41		11.69	10.45		11.55	11.10		11.18
Ship #8	11.10	11.50		11.31	11.94		12.40	10.97		11.11
Ship #9	11.99			10.85	10.07		11.76	12.59		11.90
Ship #10	11.95			10.57	11.21		10.66			11.30
Ship #11	13.65			14.09	13.81		14.52			14.85
Ship #12				14.05	13.75		14.28			15.04
Ship #13				13.77	14.48		14.54			
Ship #14				12.93	13.23					
Ship #15					14.15					
Mean:	11.58	11.30	11.79	10.95	11.50	11.67	11.63	11.85	11.36	11.38
Max:	12.15	12.09	11.94	11.69	12.91	12.08	12.62	12.59	12.10	12.14
Min:	10.72	10.41	11.50	10.28	10.07	10.86	10.66	10.97	10.82	10.32
Std Dev:	0.53	0.57	0.21	0.44	0.98	0.70	0.65	0.55	0.51	0.52
Totals:	**Mean:**	11.47	**Max:**	12.91	**Min:**	10.07	**SDev:**	0.63		

Table 4-8. Simulation Delay Times for the Redesigned Targeting Process, Pessimistic Scenario

2. Comparison Against the Current Process

Comparing the current process against the redesign immediately shows the improvements achieved using information technology. Not only has the length of the process decreased, but the handoffs and number of transcriptions have also been reduced.

Table 4-9 presents a comparison of the current and redesigned process configuration measurements. The configuration measurements were discussed and defined in Chapter III.

Configuration Measure	Current Process Value	Redesigned Process Value
Process Size	8	5
Process Length	8	3
Handoffs	3	2
Feedback loops	1	1
IT-Support	1	4
IT-Communication	0	4
IT-Automation	0	3

Table 4-9. Comparison of Configuration Measures

There are two items to note concerning the redesigned process values in Table 4-9. First, the process length is three; this number is arrived at by considering activities one through three (see Figure 4-1) as running in parallel with the two other activities in the process. The second item to note is the dramatic increase in IT support, IT communication and IT automation. These items are the result of the application of technology to the process.

To further compare the two processes the configuration measures in Table 4-9 are given to KOPeR for redesign diagnosis. The resulting diagnosis is shown in Table 4-10 along with the diagnosis for the current targeting process.

	(Numeric) - Diagnosis	
Measure Name	Current Process	Redesigned Process
Parallelism	(1.0) – sequential process	(1.667) – sequential process
Handoffs fraction	(0.375) – process friction	(0.4) – process friction
Feedback fraction	(0.125) – feedback looks OK	(0.2) – feedback looks OK
IT support fraction	(0.125) – inadequate IT support	(0.8) – IT support looks OK
IT communication fraction	(0.0) – inadequate IT communications	(0.8) – IT communication looks OK
IT automation fraction	(0.0) – IT automation first requires substantial infrastructure in terms of support and communication.	(0.6) – IT automation looks OK

Table 4-10. Comparison of Diagnoses

In the areas of parallelism, handoffs, and feedback the diagnosis does not show any difference between the current and redesigned processes. The numbers for these three measures have improved, but not to the point where KOPeR would change the diagnosis. The diagnosis for IT support, IT communication and IT automation have gone from inadequate to OK; therefore, the redesign has succeeded in eliminating three negative diagnoses.

Another area for comparison between the current and redesigned processes is with the results of the simulation runs. Comparing the numbers from the runs of the three different scenarios, the simulation model shows cycle time reductions. The current process cycle time average is 16.1 minutes per vessel. Regarding the *most likely* scenario, the cycle time is 7.6 minutes per vessel, a reduction in cycle time of 8.5 minutes, or 52.6%, per vessel. For the *optimistic* scenario, the cycle time is 4.5 minutes per vessel, a reduction in cycle time of 11.6 minutes, or 72%, per vessel. For the *pessimistic* scenario, the cycle time is 11.5 minutes per vessel, a reduction in cycle time of 4.6 minutes, or 28.6%, per vessel. A summary of the cycle times, savings and percent reductions is provided as Table 4-11.

Scenario	Cycle Time	Reduction	% Reduction	Man Years Saved
Current	16.1 min	N/A	N/A	N/A
Optimistic	4.5 min	11.6 min	72.0%	2.8
Most Likely	7.6 min	8.5 min	52.6%	2.1
Pessimistic	11.5 min	4.6 min	28.6%	1.1

Table 4-11. Summary of Cycle Time Savings by Scenario

To provide a measure of the amount of time saved annually by the redesigned process, I separately aggregate the simulation numbers from each of the three scenarios. These numbers are aggregated with the 11 ports that have large PSC programs (defined as having more than 400 examinations in a year based on the 1998 Annual PSC report). Based on simulated vessel arrivals of two to 15 vessels a day, an average number of vessel arrivals is 7.5. Assuming 7.5 vessel arrivals a day per port and a boarding ratio of 0.25, the time saved would be: 4265.9 hours per year for the most likely scenario; 5821.75 hours per year for the optimistic scenario; and 2308.6 hours per year for the pessimistic scenario. (The hours per year were calculated with the following formula:

total number hours saved yearly = minutes saved X number of vessels X (365 days in a year ÷ 60 minutes) X 11 ports) This equates to approximately 2.1 man-years saved for the most likely scenario, 2.8 man-years for the optimistic scenario, and 1.1 man-years for the pessimistic scenario, and only for the 11 ports with large PSC programs.

Comparing the critical performance measure of data accuracy of the current process to the redesigned process, the number of transcriptions has been reduced from four to one. This points to a much more accurate process which reduces, if not eliminates, the amount of rework that is required by the current process. Further, having a single point of entry for all data will strengthen data integrity of the overall system.

The number of people required to perform the redesigned process has also been reduced. Removing the watchstander, and the person performing the data entry and grading removes two people from the process. In addition to the manpower savings from cycle time reduction mentioned above, removing two personnel from the process saves approximately 1.4 man years annually. (This is calculated by taking the six minutes per vessel the two personnel would be spending in the current process and performing a calculation as shown above.) Additionally, removing the personnel frees them to perform other duties. Making the grading computerized and less prone to errors also decreases the amount of time the supervisor requires for review of work, thus freeing him to concentrate on other important issues.

If the process variation described in Section 1 above is adopted (i.e., the vessel agent calling in to a watchstander or a centralized information broker is used), there is still an improvement in cycle time and a reduction in errors, although not as dramatic as above. This improvement in cycle time is due to the removal of the data entry and vessel grading person from the process, but this is in no way a removal of the watchstander who is required to take vessel arrival information. Regarding data errors, the computerized grading would reduce them. However, the improvements in data errors would not be as great as the results seen for the full redesign, because the watchstander is still involved in the process.

The redesigned process also leverages technology better than the current process. The entire redesigned process is supported by a web-enabled database, whereas the current process relies upon multiple handwritten lists and manual evaluation. Although there are costs involved in developing, implementing and maintaining such a system, the costs associated with building from an already existing and evolving MSN would be significantly less than if the system were implemented from scratch. This is compatible with the Coast Guard development strategy of continuously evolving software as opposed to developing a static, finished product.

3. Advantages of the Proposed Process

There are numerous advantages to the proposed process over the current process. The advantages in reducing cycle time and the number of data transcriptions are the most telling, as those are the critical performance measures for the redesign to be successful. There are other advantages that can also benefit the overall organization significantly:

1. Removing additional personnel from the process by using technology frees those personnel to perform other duties at the unit.
2. Reducing the amount of time spent on rework and review allows those involved in the process more time to perform other work or hone their skills to perform the physical inspection better.
3. Having the vessel agent input the data and then receive feedback on the priority of the vessel gives the agent a better feel of what to expect for the port call. This can lead to the vessel personnel being ready for the exam and reduce the amount of time spent on board the vessel.
4. Getting the information on vessels in port and vessels scheduled to be boarded in a distributed database system allows for accurate real time statistics for analysis at the unit, district and headquarters levels.
5. The consistency afforded by having a national system for evaluating the risk of vessels entering port is invaluable. Ensuring that a vessel is evaluated the same across all ports goes a long way in building the credibility of the

program, since multiple boardings would be eliminated and vessels would be handled the same in different ports.

6. Reducing the cycle time provides the ability to raise the boarding ratio with the same number of personnel and thus reduce the chance that a ship that should be boarded will leave port without being boarded.

4. Disadvantages and Limitations

Some of the disadvantages and limitations to the redesign are as follows:

1. There is a monetary cost to the development of the redesign. The cost of implementing the redesign in reality would not be staggering, but due to the Coast Guard's budgetary limitations, cost becomes an issue.
2. The resistance vessel agents may have in changing the way they do business with the Coast Guard.
3. The resistance of ports that have centralized information brokers and their reluctance to change their business relationship with the Coast Guard.
4. The simulation model's limitation of not accounting for the degree of variability of personnel interruptions. This may change the actual cycle time reductions seen in an actual implementation of the redesign.
5. The reliance on technology for the redesign. If the ability to use the technology is compromised by a weather event etc., the process could be slowed down considerably.
6. The fact that the model parameters are estimations of reductions in cycle time may result in the estimations of cycle time savings being different than those seen in an actual implementation of the redesigned process.

B. REDESIGN OF THE VESSEL BOARDING PROCESS

1. Model and Description of the Proposed Process

Based on the information provided by KOPeR, and the premise laid out for the redesign of the targeting process, the vessel boarding process has also been redesigned.

As with the previous model above, I use the model as a guide to facilitate the discussion of the redesigned process activities. The model of the redesigned vessel boarding process is provided as Figure 4-5.

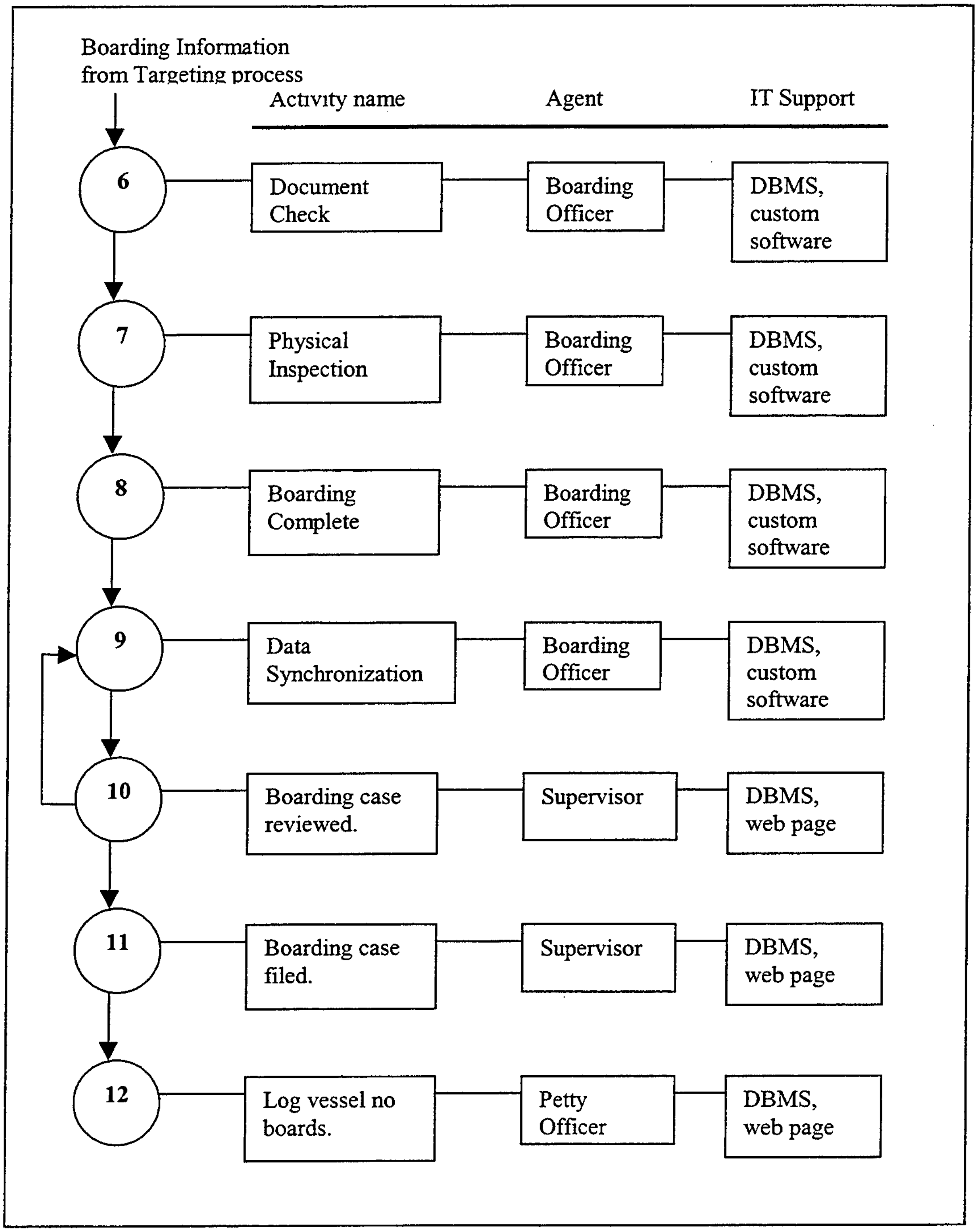


Figure 4-5. Vessel Boarding Process Redesign

The redesigned vessel boarding process begins where the targeting process leaves off. Each boarding team is equipped with a laptop computer with special software and a portable printer. The boarding teams download the pertinent vessel information from the central database to the laptop computer.

Activity node 6: Document check. On board the vessel, the vessel information is accessed on the laptop for updating. The results of the document check are input.

Activity node 7: Physical inspection. The physical inspection is conducted, and the results and documentation of the inspection are recorded on the laptop.

Activity node 8: Boarding complete. At the end of the boarding, the boarding letter, with any discrepancies, is printed and left with the Master of the vessel. If the vessel is to be detained, the appropriate letters and paperwork to detain the vessel are completed while onboard the vessel, then printed out and left with the Master.

Activity node 9: Data synchronization. Once back at the office, the updated information and boarding documentation are synchronized to the central database. This is accomplished by connecting the laptop to the office network and running a synchronization program. Once the information has been uploaded, an entry is made in the port case review log on the server for the supervisor to check the documentation of the case.

Activity node 10: Boarding case reviewed. The supervisor retrieves the case review log that consists of hyperlinks for each of the cases in question and performs the review. Any problems found are corrected on the spot, or the supervisor notifies the boarding officer of the problems requiring correction.

Activity node 11: Boarding case filed. Once the review is completed, the boarding information is locked so further alterations cannot be made. This serves as the official documentation of that particular boarding of the vessel.

Activity node 12: Log vessel "no boards". After vessels depart the port, a list of all vessels still in port is updated by a petty officer via a web page.

As with the targeting process, simulation is used for making comparisons between the current and redesigned processes. Again, any parameters, unless stated otherwise, are derived from my three years of experience in directly performing and supervising the process. Figure 4-6 is a portion of an EXTEND+BPR® simulation model covering the current vessel boarding process. As with the targeting model, a number of assumptions were made during the creation of this model to simplify the model as well as to obtain meaningful numbers for comparison between current and proposed process design. The assumptions and reasons for making them are explained during the discussion of the information flow through the simulation model.

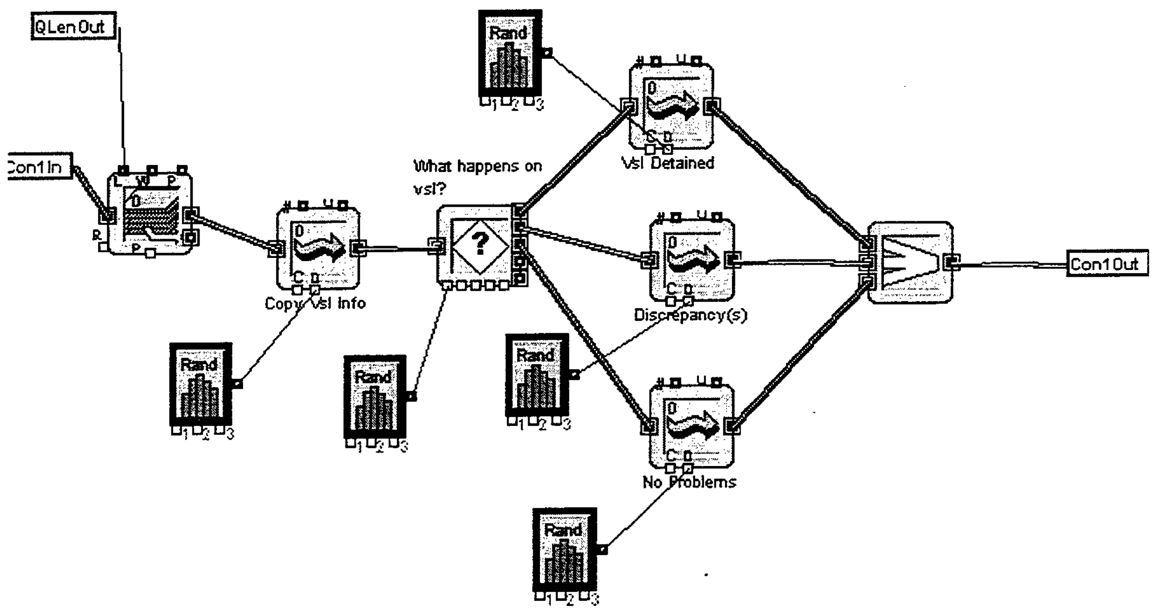


Figure 4-6. Current Vessel Boarding Process Simulation Model

Step one of this model takes the vessels that need to be boarded from the targeting process. It should be noted that the two simulation models are integrated. This means the vessels entering the vessel boarding model are those that have been selected for boarding by the targeting model. If the targeting model does not select any vessels to board, the vessel boarding model does not receive any input.

Step two is a timer block. A timer is placed at the beginning of the process to capture cycle time.

Step three is a FIFO queue. This block is a holding area for vessel information on vessels not currently being boarded. This block is used if the boarding team is assigned more than one boarding.

Step four is a transaction block. This block simulates the documentation check carried out on the vessel. The time delay for this block is assumed to be from a uniform distribution with a minimum and maximum of ten and twenty minutes per vessel, respectively. This time delay takes into account the varying amount of information about the vessel that needs to be copied.

Step four is a decision block. This determines what happens on the vessel. There are three possible outcomes for the boarding of the vessel: *no problems found*, *discrepancies found*, and *vessel detained.* The decision is made based on a random number with the probability of *vessel detained* 0.05, *discrepancies found* 0.3, and *no problems found* 0.65. These probabilities are based on three years of statistics kept at MSO Los Angeles – Long Beach (1995-1997) on vessel boardings conducted in that port.

Step five contains the three possible outcome transaction blocks: *no problems found*, *discrepancies found* and *vessel detained.* Depending upon what happens on the vessel, the information will travel to one of the three possible outcome transaction blocks. Each of the blocks has a delay, that is the amount of time it would take to account for the time spent on the vessel as well as the time required to complete documentation, review and filing time. The following assumptions apply: The delay time for the *vessel detained* block is assumed to be from a uniform distribution with a minimum and maximum of 60 and 120 minutes. This assumption is based on the amount of documentation required to perform a detention, and the extra amount of writing required when there are discrepancies that require a detention. Regarding the *discrepancies* block, the delay time is assumed to be from a uniform distribution with a minimum and maximum of 30 and 60 minutes. This assumption is based on the additional amount of writing and computer data entry required when there are discrepancies found. Regarding the *no problems* block, the delay time is assumed to be from a uniform distribution with a minimum and maximum of 20 and 45 minutes. This assumption is based on the computer entry time for the documentation of the vessel boarding.

Not pictured in Figure 4-6 is the final part of the process is the sixth step, which deals with the logging of vessels not boarded in the database. This block is fed from the targeting process and has one transaction block, which has a fixed delay time of two minutes per vessel. This assumption holds because the activity is a routine activity with non-changing information requirements.

The final assumption is identical to the final assumption of the targeting process, namely that the personnel working on the process are focused solely on completing the tasks of the process with no interruptions. Modeling of this type of uncertainty is not needed for the comparison purposes of cycle time, but is necessary to mention, as it is a possible limitation of the simulation model. The summary of parameters for this model is presented in Table 4-12.

Block Name	**Value/Range**	**Fixed/Random**	**Distribution**
Copy Vsl info	(10.0, 20.0)	random	Real, Uniform
What happens on vsl	(0.0, 1.0)	random	Real, Uniform
Vsl Detained	(60.0, 120.0)	random	Real, Uniform
Discrepancy(s)	(30.0, 60.0)	random	Real, Uniform
No Problems	(20.0, 45.0)	random	Real, Uniform
Log no boards	2	fixed	N/A

Table 4-12. Current Vessel Boarding Process Simulation Model Parameters

Measurements for this process are taken in the same manner as the targeting process described in Section B. The measurements for the simulation runs are presented in Table 4-13. Since the vessel boarding process takes the output from the targeting process, the number of vessels running through the vessel boarding model vary based on the 0.25 boarding ratio in the targeting process (the 0.25 boarding ratio phenomena was discussed in Section A.1.). Each simulation run presented in Table 4-13 corresponds with the simulation run of the same number for the targeting process, thus simulating the entire process from beginning to end.

Sim #	1	2	3	4	5	6	7	8	9	10
Ship #1	72.52	104.49	47.57	50.60				33.32	35.69	44.13
Ship #2				46.33				44.27	49.39	67.75
Ship #3										40.50
Ship #4										138.64
Ship #5										
Ship #6										
Mean:	72.52	104.49	47.57	48.47	N/A	N/A	N/A	38.80	42.54	72.75
Max:	72.52	104.49	47.57	50.60	0.00	0.00	0.00	44.27	49.39	138.64
Min:	72.52	104.49	47.57	46.33	0.00	0.00	0.00	33.32	35.69	40.50
Std Dev:	N/A	N/A	N/A	3.01	N/A	N/A	N/A	7.74	9.69	45.55
Totals:	**Mean:**	59.63	**Max:**	138.64	**Min:**	40.50	**SDev:**	30.42		

Table 4-13. Simulation Delay Times for the Current Vessel Boarding Process

The simulation model for the redesigned process is identical to the model for the current process. This is due to the structure and flows of the current and redesigned processes being identical, thus the model is unchanged. The only differences are the delay times associated with the transaction blocks that simulate what has happened on the vessel during the boarding. The times are to account for the effort spent doing the paperwork on the vessel and completing the required documentation back at the office. As for the redesigned targeting process, three scenarios were run: *most likely*, *optimistic* and *pessimistic*. As was done for the targeting process, the parameters for the most likely scenario were determined by estimating a range of the time saved for each activity. Based off the most likely scenario parameters, high and low limits for each activity were estimated, thus providing the set of parameters for the optimistic and pessimistic scenarios. Summaries of the model parameters for the three scenario simulation runs are provided as Tables 4-14, 4-15 and 4-16. The results of the simulation runs for the three scenarios of the redesigned process are presented in Tables 4-17, 4-18, and 4-19. The results were collected in the same manner as those for the current simulation model.

Block Name	Value/Range	Fixed/Random	Distribution
Copy vsl info	(10.0, 20.0)	random	Real, Uniform
What happens on vsl	(0.0, 1.0)	random	Real, Uniform
Vsl Detained	(30.0, 60.0.)	random	Real, Uniform
Discrepancy(s)	(15.0, 30.0)	random	Real, Uniform
No Problems	(10.0, 20.0)	random	Real, Uniform
Log no boards	2	fixed	N/A

Table 4-14. Redesigned Vessel Boarding Process Simulation Model Parameters, Most Likely Scenario

Block Name	Value/Range	Fixed/Random	Distribution
Copy vsl info	(5.0, 10.0)	random	Real, Uniform
What happens on vsl	(0.0, 1.0)	random	Real, Uniform
Vsl Detained	(15.0, 30.0)	random	Real, Uniform
Discrepancy(s)	(10.0, 15.0)	random	Real, Uniform
No Problems	(5.0, 20.0)	random	Real, Uniform
Log no boards	2	fixed	N/A

Table 4-15. Redesigned Vessel Boarding Process Simulation Model Parameters, Optimistic Scenario

Block Name	Value/Range	Fixed/Random	Distribution
Copy vsl info	(15.0, 25.0)	random	Real, Uniform
What happens on vsl	(0.0, 1.0)	random	Real, Uniform
Vsl Detained	(40.0, 70.0.)	random	Real, Uniform
Discrepancy(s)	(30.0, 35.0)	random	Real, Uniform
No Problems	(15.0, 30.0)	random	Real, Uniform
Log no boards	2	fixed	N/A

Table 4-16. Redesigned Vessel Boarding Process Simulation Model Parameters, Pessimistic Scenario

Sim #	1	2	3	4	5	6	7	8	9	10
Ship #1		39.46	32.32	26.55	30.90	32.51	26.96	25.90	53.10	30.40
Ship #2		53.46	69.80		44.09		41.34	53.96		
Ship #3		29.61	35.71				30.88	34.29		
Ship #4							36.70	39.47		
Ship #5								29.58		
Ship #6								43.85		
Ship #7										
Mean:	N/A	40.84	45.95	26.55	37.49	32.51	33.97	38.40	53.10	30.40
Max:	N/A	53.46	69.80	26.55	44.09	32.51	41.34	53.96	53.10	30.40
Min:	N/A	29.61	32.32	26.55	30.90	32.51	26.96	25.90	53.10	30.40
Std Dev:	N/A	N/A	20.73	N/A	9.33	N/A	6.34	N/A	N/A	N/A
Totals:	**Mean:**	38.37	**Max:**	69.80	**Min:**	29.58	**SDev:**	11.46		

Table 4-17. Simulation Delay Times for the Redesigned Vessel Boarding Process, Most Likely Scenario

Sim #	1	2	3	4	5	6	7	8	9	10
Ship #1	24.73	13.35		17.51		20.96	18.76		14.02	26.81
Ship #2		24.76		29.23			28.86			25.34
Ship #3		17.79		28.11						31.85
Ship #4				36.13						
Ship #5				16.18						
Ship #6										
Ship #7										
Mean:	24.73	18.63	N/A	27.75	N/A	20.96	23.81	N/A	14.02	28.00
Max:	24.73	24.76	N/A	36.13	N/A	20.96	28.86	N/A	14.02	31.85
Min:	24.73	13.35	N/A	17.51	N/A	20.96	18.76	N/A	14.02	25.34
Std Dev:	N/A	5.75	N/A	7.69	N/A	N/A	7.14	N/A	N/A	3.41
Totals:	**Mean:**	23.88	**Max:**	36.13	**Min:**	16.18	**SDev:**	6.64		

Table 4-18. Simulation Delay Times for the Redesigned Vessel Boarding Process, Optimistic Scenario

Sim #	1	2	3	4	5	6	7	8	9	10
Ship #1	49.51	39.74	46.23	35.54	43.91	41.20	71.07	42.54	45.94	30.40
Ship #2	65.29	60.30		64.57	70.43			92.48		
Ship #3	51.25			46.17	48.81			50.03		
Ship #4				60.87	65.20					
Ship #5										
Ship #6										
Ship #7										
Mean:	N/A	50.02	46.23	51.79	57.09	41.20	71.07	61.68	45.94	30.40
Max:	65.29	60.30	46.23	64.57	70.43	41.20	71.07	92.48	45.94	30.40
Min:	49.51	39.74	46.23	35.54	43.91	41.20	71.07	42.54	45.94	30.40
Std Dev:	8.65	14.54	N/A	13.43	12.73	N/A	N/A	26.94	N/A	N/A
Totals:	**Mean:**	53.40	**Max:**	92.48	**Min:**	46.17	**SDev:**	14.56		

Table 4-19. Simulation Delay Times for the Redesigned Vessel Boarding Process, Pessimistic Scenario

2. Comparison Against the Current Process

The redesigned process looks much like the current process with the exception of the addition of technology to remove and streamline some activities. While the changes to the process are not as radical as those proposed for the targeting process, the changes nevertheless promise to improve the process significantly by using automation to remove redundant copying of data and removing lost time in the documentation activities.

The redesigned process has lost the filing activity the current process has, since all of the official documentation is now kept electronically. This makes the process one activity shorter than the current process.

Table 4-20 presents a comparison of the current and redesigned process configuration measurements. The configuration measurements were discussed and defined in Chapter III.

Configuration Measure	Current Process Value	Redesigned Process Value
Process Size	8	6
Process Length	8	6
Handoffs	3	1
Feedback loops	1	1
IT-Support	2	6
IT-Communication	0	6
IT-Automation	0	3

Table 4-20. Comparison of Configuration Measures

There are two items to note concerning the redesigned process values in Table 4-20. First, the process length is six; this is the result of removing the compilation of the vessel boarding documentation activity of the current process (see Figure 4-5) and considering the final activity a process unto itself. The second item to note is the dramatic increase in IT support, IT communication and IT automation. These items are the result of the application of technology to the process.

To further compare the two processes, as done for the targeting process, the configuration measures in Table 4-20 are given to KOPeR for redesign diagnosis. The resulting diagnosis is shown in Table 4-21 along with the diagnosis for the current vessel boarding process.

	(Numeric) - Diagnosis	
Measure Name	Current Process	Redesigned Process
Parallelism	(1.0) – sequential process	(1.0) – sequential process
Handoffs fraction	(0.375) – process friction	(0.167) – handoffs look OK
Feedback fraction	(0.125) – feedback looks OK	(0.167) – feedback looks OK
IT support fraction	(0.25) – inadequate IT support	(1.0) – IT support looks OK
IT communication fraction	(0.0) – inadequate IT communications	(1.0) – IT communication looks OK
IT automation fraction	(0.0) – IT automation first requires substantial infrastructure in terms of support and communication.	(0.5) – IT automation looks OK

Table 4-21. Comparison of Diagnoses

In the areas of parallelism and feedback, the diagnosis does not show any difference between the current and redesigned processes. Parallelism shows no improvement, this is due to the decision, based on arguments made in Chapter III, to leave the process sequential. Feedback shows no difference because the number of feedback loops in the process were not changed. The diagnosis for handoffs has gone from process friction to OK. The diagnosis for IT support, IT communication and IT automation have gone from inadequate to OK. Therefore, the redesign has succeeded in eliminating these last four negative diagnoses.

The largest differences from the current process lie in the area of the critical performance measures for the redesign. The average simulation cycle time for the current process is 59.6 minutes, whereas the *most likely* scenario, has a cycle time of 38.4 minutes per vessel. This constitutes a reduction in cycle time of 21.2 minutes per vessel, a reduction of 35.6%. For the *optimistic* scenario, the cycle time is 23.9 minutes per vessel. This reduces the cycle time by 35.7 minutes per vessel, a 59.9% reduction. For the *pessimistic* scenario, the cycle time is 53.4 minutes per vessel. This is a slight decrease in cycle time of 6.2 minutes per vessel, or a 10.4% reduction. A summary of the cycle times, savings and percent reductions is provided as Table 4-22.

Scenario	Cycle Time	Reduction	% Reduction	Man Years Saved
Current	59.6 min	N/A	N/A	N/A
Optimistic	23.9 min	35.7 min	59.9%	2.3
Most Likely	38.4 min	21.2 min	35.6%	1.4
Pessimistic	53.4 min	6.2 min	10.4%	0.4

Table 4-22. Summary of Cycle Time Savings by Scenario

As shown with the vessel targeting process, aggregating the numbers over the 11 ports that have large PSC programs provides an idea of the amount of time saved annually by the redesigned process. Based on the simulation, the average number of vessel boardings a day per port is two. Assuming 2 vessel boardings a day per port and a boarding ratio of 0.25, the time saved would be: 2837.3 hours per year for the most likely scenario; 4777.9 hours per year for the optimistic scenario; and 829.8 hours per year for the pessimistic scenario. (The hours per year were calculated with the following formula: total number hours saved yearly = minutes saved X number of vessels X (365 days in a year ÷ 60 minutes) X 11 ports) This equates to approximately 1.4 man-years saved for the most likely scenario, 2.3 man-years for the optimistic scenario, and 0.4 man-years for the pessimistic scenario, and only for the 11 ports with large PSC programs.

With respect to the critical performance measure of data transcription, the number of times data are transcribed for the redesigned process is zero compared to four for the current process. This is a particularly telling number in that there is no information being copied repeatedly as it is in the current process. Thus, the data going into the process will be more accurate and timely. The combination of the number of transcriptions, and the fact that the documentation is done onboard the vessel, as opposed to doing the documentation later, leads to a much improved process in terms of accuracy and timeliness.

In the area of technology, the redesigned process is much more technology enabled than the current process. Where the current process requires pen and pencil record keeping and rudimentary use of a database, the redesigned process relies on web pages, a database and electronic record keeping. Documenting the boarding in the current process is a task that is not completed at one time; rather, part is done onboard the vessel, and the final documentation is done in the office. The completion of the documentation can, at times, take place two or three days after the vessel boarding, especially if the vessel is detained. In the redesigned process, the documentation of the boarding is completed onboard the vessel. This includes any required paperwork to detain the vessel.

3. Advantages of the Proposed Process

The redesigned process has many advantages over the current process. These advantages are as follows:

1. The reductions in cycle time and number of transcriptions, which were shown in the previous section, are the most significant. Reducing the cycle time on the process allows for more timely documentation of the boarding, which

leads to the elimination of redundant boardings by other MSOs. Additionally, the time saved allows for a reduction in manpower, as fewer boarding teams need to be fielded to perform the same number of vessel boardings.

2. The improvement in accuracy by reducing the number of transcriptions allows for less rework and time spent by the supervisor in reviewing the case work. This gives the supervisor more time to perform other duties. The boarding officer also benefits from the improved accuracy by not having to rework errors made in the case. Overall, the morale of the personnel performing the process will likely improve, because there will be less time spent on the paperwork and more time spent on training and doing the job at hand.
3. Completing the documentation on a laptop computer and synchronizing with the main database provides immediate access to the documentation of the vessel boarding. With the documentation available immediately, any questions on what happened on the boarding can be immediately accessed without having to shuffle through papers or hunt down the boarding officer for details.
4. Having a nationwide standard for the completion of vessel boarding documentation reduces the "learning curve" for boarding officers reassigned to another unit, as the process will be the same everywhere. Additionally, the consistency of the documentation will allow for direct comparisons of unit performance of the PSC process.
5. The filing of the case work electronically saves on office space for filing cabinets, and reduces the burden of keeping track of the archival process for federal records. Since the official records are kept on the database server at a centralized location, the use of regular backups and automated archival allows this process to be performed in a single location with a large reduction in manpower to complete the process.

4. Disadvantages and Limitations

There are a number of disadvantages and limitations to the redesigned vessel boarding process, they are as follows:

1. The large cost involved in providing portable computers and printers to all MSOs that perform the PSC process.
2. The cost of developing the software to implement the system, while not staggering, is a concern considering the Coast Guards limited budget.
3. The simulation model's limitation of not accounting for the degree of variability of personnel interruptions. This may change the actual cycle time reductions seen in an actual implementation of the redesign.
4. There may be resistance among the boarding officers in using portable computers to document the vessel boardings; especially among the older personnel who are not as comfortable using computers.
5. The loss of a portable computer would result in the loss of the entire record of the vessel boarding. While not a common occurrence, the loss of equipment while boarding a vessel off shore does occur.
6. There may be resistance, by personnel who want to keep paper records of the boardings locally. This would negate the savings of office space for filing cabinets.

C. MERGING OF THE TWO PROCESSES

Since the entire process was split into two separate parts, the targeting process and the vessel boarding process, for the redesign, the integration of the two parts needs to be addressed. The links between the two processes are the list of the vessels identified for boarding and the list of vessels due in port. These two pieces of information are required for the vessel boarding process to start and complete.

While partially addressed by the description of the redesigns, it is important to understand this linkage. At the completion of the targeting process, there are two lists of vessels; one is the list of vessels to be boarded, and the other is the list of vessels due into port. Both of these lists are stored on the database server for each port in the nation. When a boarding team has been assigned a vessel to board, the boarding officer uses a laptop computer to load the pertinent vessel information into a local database on the computer. This is the linkage between the two processes.

For the entire PSC process to be complete, the list of vessels in port has to be updated when vessels leave the port. Since this list is stored on the database server, it is only a matter of accessing the list on a web page, and updating the status of the vessels. This completes the PSC process.

D. SUMMARY

In this chapter, two processes, the targeting process and the vessel boarding process, have been redesigned based on the recommendations of Chapter III. In addition to the redesign, simulation was used to compare the possible reductions in critical performance measures between the current and proposed processes using three scenarios; most likely, optimistic and pessimistic.

In the next chapter, a proof of concept prototype is developed to test the technology of the redesign in this chapter. It consists of the development of a database, a decision support function, web pages to support interaction with the database, and an application to support the remote use of the database on board a vessel.

V. IMPLEMENTATION

This chapter describes the development and implementation of a proof of concept prototype of the redesigned process. To see how the redesigned process will actually operate, a prototype implementation of the process has been developed to establish proof of concept. Table 5-1 provides a listing of the tools used in creating and running the prototype called RePortS (for Re-engineered Port System). The actual implementation of the full system, by a team of professional software engineers, will be more polished and use more "industrial strength" tools than those used here.

Category	Tool Name	Purpose
Development and Deployment tools	Flow Charting PDQ Lite 1.1j	Create Data Flow Diagrams
	SALSA	Create Semantic Object Model
	Microsoft Access 97	Provide the back end database and portable application and database
	Evrsoft 1st Page 2000	Build the web pages for the web based portion of the application
	Microsoft Internet Information Server 4.0 (IIS)	Provide HTTP service and active server page rendering of the web pages and data from the database.
	Microsoft Internet Explorer 5.0	Used to test the application.

Table 5-1. Tools Used to Implement RePortS

To assist in the completion of RePortS, a Rapid Application Development (RAD) systems development methodology is used. This approach provides a quick way to develop a system compared to traditional methods (Hoffer 1999, p.492), and therefore is particularly appropriate for prototype development. The RAD life cycle has four phases: requirements, design, construction and roll out.

A. REPORTS PROTOTYPE REQUIREMENTS AND DESIGN

The first step of the process is to identify the requirements for RePortS. To this end, the requirements for the system are derived from the description of the process provided in Chapter IV. These requirements in broad terms are as follow:

1. The system will have a database.
2. The database will be accessible via the Internet/Intranet.
3. The system will interact with external entities as well as entities internal to the Coast Guard.
4. The system will have a decision support function.
5. There will be a capability to update the main database from a portable database.
6. Hard copy documentation will be created from a portable computer as well as a computer connected to the Internet/Intranet.
7. All review functions will occur online.

While these requirements are general in nature, the intent of the RAD approach is to allow the user to assist in the development process and identify requirements as the development proceeds. Since the user and developer of the system are the same person in this case, the identification of additional requirements proceeds with the development of the system. It should be noted that the primary motivation for development of RePortS is to show that the redesign of the process is feasible, not to fully implement all features that an actual implementation of the process would require.

The next step of the RAD approach is design. In the design phase, the requirements and user input are transformed into a data model and a process model. These models, shown below, not only document the requirements, but also assist the developer to understand the requirements better.

1. Database Design: Conceptual Model

The technique used to create the data model is the semantic object model (SOM). The SOM uses semantic objects to represent identifiable things in the users' work environment. More formally, a semantic object is *a named collection of attributes that sufficiently describes a distinct identity.* (Kroenke 1998, p72) For this prototype, the original intention was to use the data model for the MSN project. However, it has not been possible to acquire a copy, or even to ascertain if one actually exists. Therefore, I have created a SOM based on the information needed to successfully build the prototype as shown in Figure 5-1.

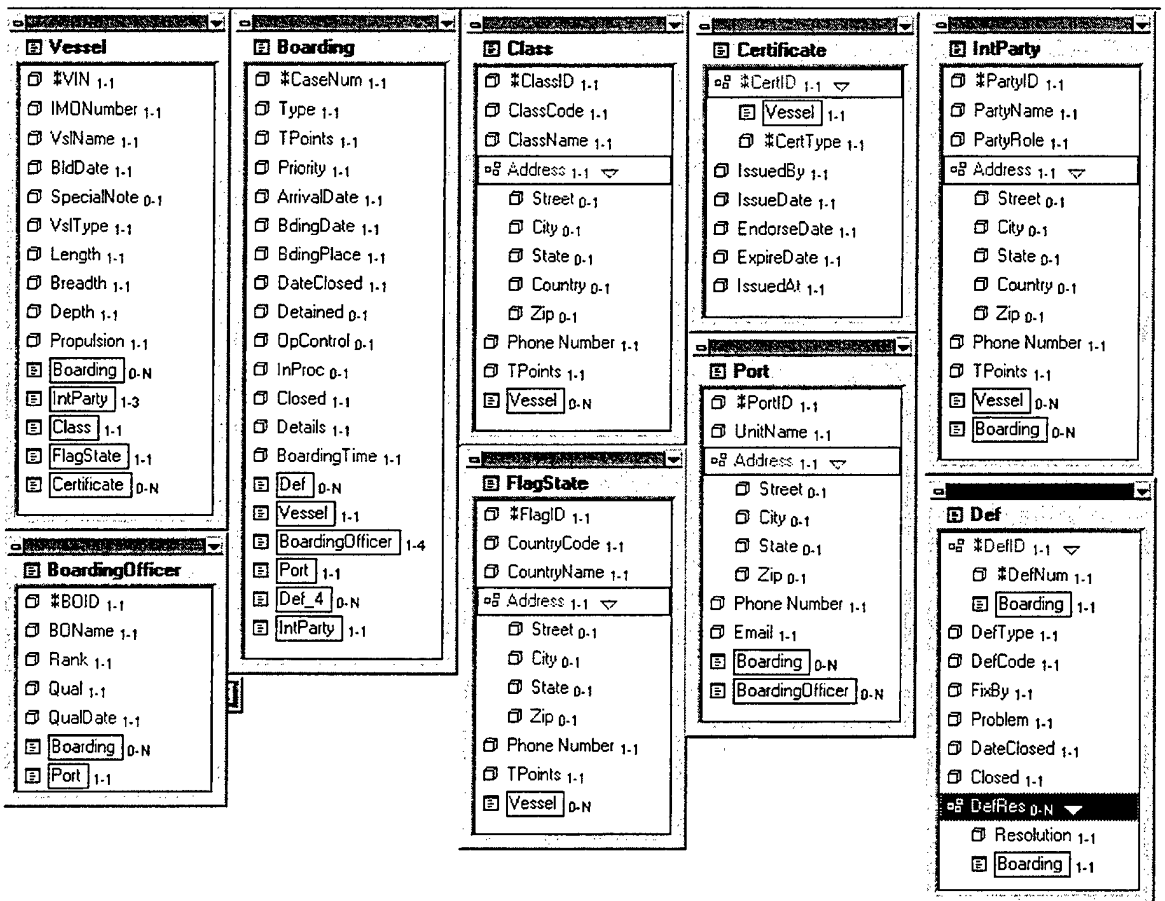


Figure 5-1. Semantic Object Model for RePortS

The conventions used in the SOM by Salsa are explained below. Each object is shown as a rectangle with the name of the object on the top. Each object has a list of attributes listed in the window of the object. The key field(s) of an object is (are) denoted by two asterisks stacked on top of each other. Attribute cardinalities are given next to each attribute, (e.g., $VslName_{1-1}$, where the $_{1-1}$ is the cardinality of the attribute VslName). The cardinality is denoted as minimum and maximum cardinality from left to right, so in the above example the minimum number of instances of VslName in the record is one, and the maximum instance is also one. To represent a *many* relationship, the symbol "N" is used. A table's relationship with another table is denoted by listing the related table name in the list of attributes with a box around it, (i.e. $\boxed{\text{Boarding}}_{0-N}$) with the cardinalities used in the same manner as described above.

Each of the objects in the SOM is defined and described:

1. The Vessel object is designed to capture the data pertinent to a vessel. It has the attributes that describe a vessel which can have a boarding done on it. A vessel can have up to three Interested Parties; the owner, operator, and agent. Additionally, a vessel can have only one Class and Flag State, but may have many Certificates.
2. The Boarding object is used to capture the information from a physical inspection of a vessel. It captures the where, what kind, when, etc. dimensions of the boarding. Additionally, it captures the grading information on a vessel for a port call. A boarding may have many deficiencies (represented by a Def object), and may only be done on one vessel. A boarding is performed by one to four boarding officers in only one port. A boarding also has only one interested party who is the agent for the vessel.
3. The Def object is a deficiency found during a particular boarding on a particular vessel. It contains the information needed to describe the nature of the discrepancy as well as the requirements needed to repair the problem and resolve the discrepancy (i.e., have it marked as *closed* in the database). The deficiency is only identified on one boarding and when cleared, it becomes related to the boarding that cleared the deficiency.
4. The Certificate object captures the information contained on the certificates issued to a vessel. This object has issue, endorse, and expiration dates, as well as who issued the certificate. Each certificate is issued only to one vessel.

5. The Class object represents the classification society of the vessel. The name of the classification society, contact information and targeting points are the items captured by this object. A classification society can class many different vessels.
6. The IntParty object represents the owners, operators, agents, and any other party that has an interest in the vessel. Like the Class object, it contains the contact information for the party it represents, and the targeting points for the particular party. An interested party can be associated with many different vessels and boardings.
7. The FlagState object represents the country of registry of the vessel. It contains the contact information for the Flag State's maritime attaché, and the targeting points for each Flag State. A Flag State can have many vessels in its vessel registry.
8. The Port object represents the Coast Guard unit covering a particular port area. It contains the name of the unit and contact information. A port has many boarding officers, and many boardings may be conducted in the port.
9. The BoardingOfficer object represents the boarding officers performing boardings on the vessels. The name of the boarding officer, qualifications, current unit, and other identifying information are contained in the BoardingOfficer object. A boarding officer performs many boardings, but is associated with only one port.

2. Database Design: Relational Model

The next step in database design is to transform the SOM into relational tables in a database, which requires a three-step procedure.

1. Transform the object diagrams into relations. The relations take the form of the relation name with the key field of the relation followed by the attributes.
2. Normalize relations to remove modification anomalies, such as insertion and deletion anomalies.

3. Transform the normalized relations into a table definition, which can be incorporated into the database schema. The schema defines the database's structure, tables, relationships, domains and business rules. (Kroenke 1998, p. 30) From the schema, the physical database can then be created.

Based on the SOM in Figure 5-1, the data model is transformed into relations by using the transformation methodology suggested by Kroenke. (Kroenke 1998, pp. 163-185) In this methodology, each semantic object is mapped into one or more relations. Relational notation is different from that for the SOM. The key field for each table is underlined and foreign keys from other tables, that are used to create a relationship with those tables, are denoted with a "_FK" after the name of the attribute. Table 5-2 shows the relations created by this process.

With the transformation complete, the resulting relations need to be normalized to prevent any relational anomalies, discussed previously, from occurring. The classes of relations, and the techniques for preventing the anomalies are called normal forms. (Kroenke 1998, p.117) There are several classes of normal form; however, this implementation will be normalized only to second normal form. Using second normal form affords a good tradeoff between anomaly reduction and simplicity in relational database design. This is an appropriate balance for a prototype that will not be used in a production environment where a more normalized form would provide benefits from a greater reduction of deletion anomalies. A relation is in second normal form when all of its non-key attributes are functionally dependent on *all*, and not just a subset of, the key attributes. Another test of second normal form is to have single attribute keys. (Kroenke 1998, p.118) Analyzing the relations in Table 5-2 reveals the relations are already in second normal form. All but five of the relations have single key attributes, two relations with multiple attribute keys have no other attributes, and in the other relations, each non-key attribute depends upon all of the key attributes (in order to access any of the attributes, both parts of the relation key are needed).

Table	Attributes
VESSEL	Vin, IMONumber, VslName, SpecialNote, VslType, BldDate, Length, Breadth, Depth, Propulsion, ClassID_FK, FlagID_FK
BOARDING	CaseNum, Type, ArrivalDate, Tpoints, Priority, InProc, BdngDate, BdngPlace, Detained, OpControl, DateClosed, Closed, Details, BoardingTime, Vin_FK, PortID_FK, PartyID_FK
CLASS	ClassID, ClassCode, ClassName, Street, City, State, Country, Zip, TPoints, PhoneNo
INTPARTY	PartyID, PartyName, PartyRole, Street, City, State, Country, Zip, TPoints, PhoneNo
DEF	DefID, CaseNum_FK, DefType, DefCode, FixBy, Problem, DateClosed, Closed
FLAGSTATE	FlagID, CountryCode, CountryName, Street, City, State, Zip, TPoints, PhoneNo
VESSEL-INTPARTY	Vin_FK, PartyID_FK
CERTIFICATE	CertType, Vin_FK, IssuedBy, IsuueDate, EndorseDate, ExpireDate, IssuedAt
BOARDING OFFICER	BOID, BOName, Rank, Qual, QualDate, PortID_FK
PORT	PortID, UnitName, Street, City, State, Zip, PhoneNo, Email
BOARDING–BOARDING OFFICER	CaseNum_FK, BOID_FK
DEFRES	DefID_FK, CaseNum_FK, Resolution, CaseNumRes_FK

Table 5-2. RePortS Relations

The table definition now follows the transformation to relations and normalization. Appendix B contains the table definition for the relations presented in Table 5-2. The table definition contains the table names, their attributes, key and foreign key fields, and the type and size of all attributes. With the information contained in the table definition and relations, there is enough information in the schema to create the database.

An area not represented in the database schema is that of the business rules. Business rules specify the constraints on allowed data values that must be enforced. These constraints are enforced by the database and the applications that interact with the database and/or the users of the database. (Kroenke 1998, p.32) A large number of the rules for data are captured in the SOM, for example, a vessel can have one and only one Flag State. The business rules not identified by cardinalities or in other areas of the schema are identified as follows:

1. When a vessel arrival is entered, a boarding is created for the vessel arrival.
2. Using any type of boarding can clear deficiencies.

These business rules will be enforced in the application interface with the database.

3. Process Design: Data Flow Diagram (DFD)

Another model commonly used to assist in the development of systems is the data flow diagram (DFD). A DFD provides a picture of the movement of data between external entities and the processes and data stores within a system. (Hoffer 1999, p278) DFDs are composed from four different symbols: the process, data store, source/sink and data flow. A sample of each type of symbol is shown in Figure 5-2.

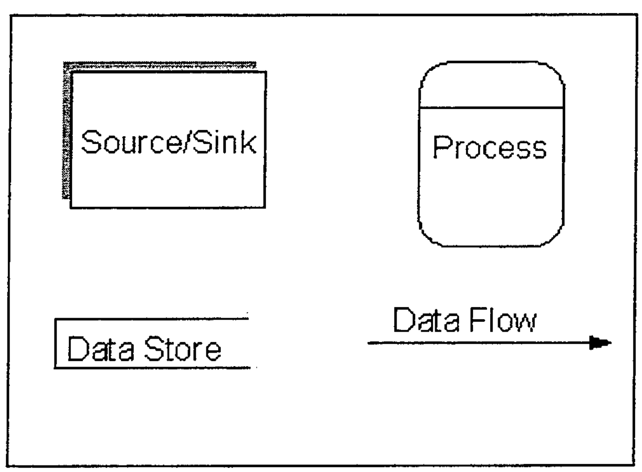


Figure 5-2. DFD Symbols

An explanation of each of the symbols is provided for those unfamiliar with the DFD modeling approach. The process is work or action performed on data to transform, store or distribute it. The data store is data at rest; it is stored in some sort of media, be it a folder or a database. The source/sink is the origin or destination of the data, often referred to as an external entity. The data flow is the movement of data between two points in the system. Typically, the data flow is labeled with the type of data moving along that data flow. (Hoffer 1999, pp.280-281)

A DFD for RePortS (Figure 5-3) has been built to better identify the flows of data and the different processes needed to implement the prototype. The DFD shows the different entities that interact with the system, the data stores in the system and the different processes that act on the data.

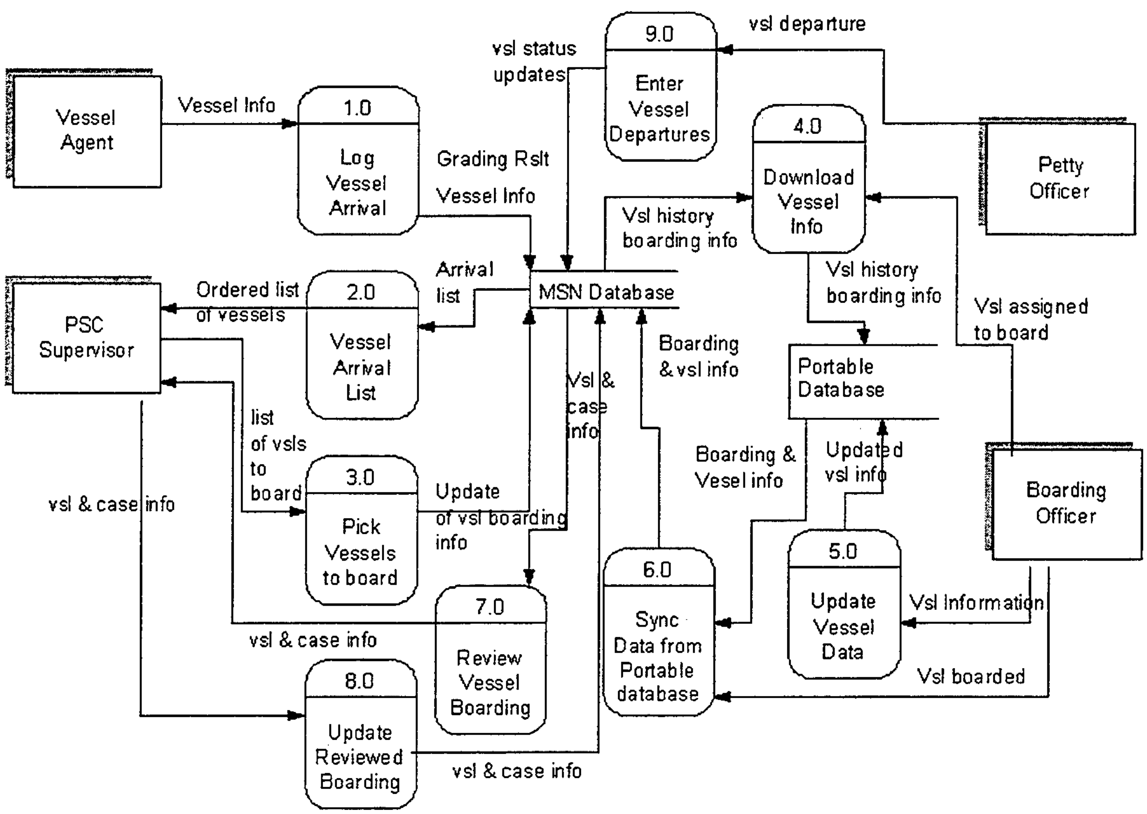


Figure 5-3. DFD for the RePortS Prototype

As seen in Figure 5-3, there are nine distinct processes that act on the data either provided by the sources/sinks or drawn from the data stores. These nine processes will be the focus of the prototype development efforts. All but one of the processes represent web pages and code associated with the transformation and delivery of data from the data stores. The other process, 5.0 Update Vessel Data, represents the interface on the portable database for use onboard a vessel. It is vital to understand the data flows into and out of each of the processes; therefore, each of the processes is described along with its respective inputs and outputs:

1. The Log Vessel Arrival process is the transformation of the vessel arrival information provided by the Vessel Agent into a vessel boarding and grading result, which then gets stored in the MSN data store.
2. The Vessel Arrival List process queries the MSN data store for the vessel arrivals for a particular port. The process then rank orders the list and presents it to the Supervisor for his action.
3. The Pick Vessels to Board process takes the input from the Supervisor and updates the vessels selected for boarding and stores the updated information in the MSN data store.
4. The Download Vessel Info process queries the MSN data store for a particular vessel and boarding for the port call which is provided by the Boarding Officer. Then the resulting data is placed into the portable data store by the process.
5. The Update Vessel Data process takes updated vessel data and the results of the boarding from the Boarding Officer and posts the information to the portable data store.
6. The Sync Data from Portable Database process, with vessel information provided by the Boarding Officer, queries the portable data store for the updated vessel and vessel boarding information and then updates the appropriate records in the MSN data store.

7. The Review Vessel Boarding process queries the MSN data store for the vessel boarding cases that are outstanding and presents a list to the Supervisor. Based on the input from the Supervisor, the process then provides a single boarding case and vessel information package for the Supervisor to review.
8. The Update Reviewed Boarding process, takes the results of the review (any updates to the case and if validated or not) from the Supervisor and then updates the MSN data store.
9. The Enter Vessel Departures process provides a list of vessels still in port, which the Petty Officer can then update for those vessels that have departed the port. The process then updates those records in the MSN data store.

The DFD and its description provide the design information needed to develop the software functionality for the prototype. This coupled with the database schema provides adequate design information to begin the construction of the software functionality portion of RePortS. The final parts of the design process are to identify the hardware and user interface portions of the prototype.

4. Physical Design

The hardware for RePortS consists of a server computer running Microsoft Windows NT Workstation 4.0 with Microsoft Internet Information Server (IIS) 4.0 installed to provide the Hypertext Transfer Protocol (HTTP) service and active server page (ASP) rendering. Since MSN is designed to work over the CGWEB, the client machine is connected to the server by a network using 10Base-T Ethernet. The client machine, running Microsoft Windows 95 OSR2, uses Microsoft Internet Explorer 5.01 to access the web pages from the server. Additionally, the client machine is a portable computer, with Microsoft Access 97 installed to provide the portable database application for use on board a vessel.

Since the majority of the application will consist of web pages, an explanation of the underlying technology of IIS and active server pages is in order. IIS is "a network file and application server for the Microsoft Windows NT Server 4.0 operating system." (Microsoft Press 1998, p. 2) IIS provides many standard Internet services; however, the one of major importance here is the World Wide Web (WWW) service. The WWW service supports HTTP 1.1, which allows the publishing of content to the Internet. IIS has features that allow the creation and deployment of Web-based applications. Microsoft calls these features web server extensions; the two that are used in RePortS are ASP and Open Database Connectivity (ODBC), which provide interactive and dynamic access to the database on the server. ASP provides the ability to embed scripts in a web page, which are executed on the server before the web page is served to the client. ASP supports the use of scripting languages such as VBScript, JScript, and Perl. VBScript is used for the scripting language in RePortS, largely because of the convenience of familiarity and because it is easy to learn. ODBC allows the scripts on the web page to connect to the database on the server and allows the use of Structured Query Language (SQL) to extract data from and write data to the database. The combination of these two features forms the primary software technology used in implementing RePortS. The other technology utilized for the portable database application on the vessel consists of an application built in Microsoft Access 97. The database on the portable will mirror the database on the server, but will only contain the information needed to complete the boarding documentation and vessel information updates.

The final activity in the design process is the user interface. For the majority of the application, the web browser is the primary interface. The web pages all have the same graphical look to make them a coherent application. They employ lists, forms, text boxes, check boxes, combo boxes and hyperlinks to provide the interaction with the user. The interface for the portable database is in the form of an application built in Microsoft Access 97. This interface consists of a graphical menu system and forms for completing the updating of the vessel and boarding data. Screen shots of the various pages and forms are presented later in the roll out phase of the design methodology.

B. PROTOTYPE CONSTRUCTION

The next step in the RAD development methodology is construction. Construction takes the requirement and design information and generates a software solution. The construction of RePortS has three distinct activities; database creation, web application development, and portable database application interface.

1. Database Creation

The first activity, database creation, takes the database schema created in Section A. and creates the tables and relations in the database. For RePortS, the database is created in Microsoft Access 97. The procedure for completing this is as follows. Tables are created and data members are entered in the design window for the table; extensive use of drop down boxes and other visual aids make this process proceed quickly. Primary keys are identified for the tables and are noted with a key icon in the design window of the table. Once the tables are created, the relations have to be created in the relationship window. This is accomplished by using a drag and drop method of dragging the key field of one table to the related field of another and then filling in the form that pops up when the mouse button is let up. Upon completion of the table and relationship creation process, the database is ready to use. The tables are filled in with some fictitious but realistic data to allow demonstration of RePortS at roll out.

2. Web Application Development

The second activity, web application development, is a more complicated undertaking. This is due to the amount of code and application logic that needs to be created for RePortS to work. As mentioned in Section A, the technology used in creating the web application is ASP with VBScript and an ODBC connection to the database. There are three basic building blocks used in the creation of the web pages for the web application:

1. the connection to the database;
2. an SQL query that is passed to the database via the connection;
3. the manipulation of the data.

To facilitate the discussion of the three building blocks, a portion of a web page developed for RePortS is provided in Table 5-3. Line numbers have been added to assist in identifying the parts of the code.

```
<%
SQLAGENT="SELECT PartyID, PartyName FROM INTPARTY WHERE PartyRole = 'agent'"
set connagent = server.createobject("ADODB.Connection")
connagent.open "test2"
set agents=connagent.execute(SQLAGENT)
%>
<select name="AgentID">
            <% do while not agents.eof %>
              <Option value = "<%= agents(0) %>"> <%= agents(1) %></Option>
            <%agents.movenext
            loop%>
</select>
<% connagent.close %>
```

Table 5-3. Web Page Code

Before covering the three basic building blocks, the structure of the code is described for those unfamiliar with ASP. ASP code is usually contained in Hypertext Mark up Language (HTML) documents. HTML uses tags to provide instructions to web browsers on how to display a document. ASP uses specific tags to indicate to the web server that the items contained within the tags are script to be processed before delivering the page to the web browser. The tags used for ASP are identified as '<% %>' with the code to be run inside of the tags. Once the server has processed the code, the result is a document that has been dynamically built and delivered to the client. The code in Table 5-3 is used in RePortS to create a drop down list box populated with items for inclusion on another web page. With this background covered, the next step is to describe the three basic building blocks used in the creation of RePortS.

a. Connection to the Database

The first of the basic building blocks of the web application development activity is the connection to the database. The connection to the database is accomplished using ActiveX Data Objects (ADO), a technology developed by Microsoft to access databases through ODBC. In order to make the connection to the database on the server, the database must be registered with a system data source name (DSN) in the ODBC applet that is found in the control panel of the server computer. A connection can then be established via ADO in code in the web page. Line number three in Table 5-3 shows the creation of the data object on the server. Creating the object is done by calling the 'createobject' method of the server object with the parameter of 'ADODB.Connection', and then assigning the result into a variable called 'connagent'. The next step is to open the connection, as shown in line four, and is achieved by calling the 'open' method of the just created connection object and passing the DSN of the database for which the connection is desired. The connection to the database is then ready to be used to communicate with the database. After the connection has been used for the last time, it is closed, as shown in line 13. Calling the 'close' method of the connection object closes the connection.

b. SQL Query

The second basic building block of the web application development activity is the SQL query. The retrieval, updating and deleting of data in the database is accomplished using SQL queries. The query is put into a variable and then passed to the database using the connection object. In the example in Table 5-3, the query is sent to the database in line number five by calling the 'execute' method of the connection object with the query variable as the parameter. The results are returned as a record set and saved for further processing in a variable. The query in this example is a simple query, which does not contain any other variables. Many of the queries in the application use variables that are passed from page to page and query multiple tables in the database.

c. Manipulation of Data

The third basic building block of the web application development activity is the manipulation of the data. This is the stage where the data provided by the query through the connection to the database is used, modified or updated. In the example in Table 5-3, lines seven through twelve show simple manipulation of the record set provided by the query. The record set also has methods that allow the movement of a cursor to the different rows in the record set. There are also methods used to determine the placement of the cursor (e.g., beginning of file [bof] or end of file [eof]). In the example, the 'movenext' method is called to move the cursor to the next record in the record set. The columns of the record set are accessed by using subscripts with the first column being zero. In the example, the code creates a drop down list box filled with data for use in another page. To understand how this is done each line will be explained:

1. Line 7 is an HTML tag, which assigns a unique identifier to the drop down list box; this is for identifying the selection of the drop down list box when the information is sent to the server.
2. Line 8 is a do loop which tests to see if the end of the record set has been reached, and if not, the items after the statement are executed.
3. Line 9 is the assignment of the data to the list of the drop down list box. The first column is assigned to the value to be passed and the second column is the information the user will see when the list box is opened.
4. Line 10 moves the cursor to the next record
5. Line 11 causes the loop to move execution back to line eight.

Once the end of the file is reached, the drop down list box has been populated with data and is ready for use. Other more complicated manipulations of data are used in RePortS, however the concept of the manipulation is the same.

d. Summary of Basic Building Blocks

This subsection has shown the basic building blocks used in the web application development. Every ASP page used in the application uses some form of each of the blocks to carry out the functionality of the page. Although some implementations might be more complex than the example shown in Table 5-3, the basic ideas behind them are the same.

3. Portable Database Application Interface

The third activity is the portable database application interface. The portable database is a copy of the server database without all of the data the server database contains. The interface for this database is built in the Access 97 application itself. Using the wizards to build the data entry forms, queries, reports and menu navigation system is a point and click process. After the wizard has created the required objects, they are cleaned up and graphics are added to create a coherent application.

C. ROLL OUT

The final phase of the RAD development cycle is the roll out of the product. If the RAD development cycle were to be used for an actual production system, the design and construction phases would cycle a few times so the users of the system could provide feedback to the designers. In the case of RePortS, several iterations have occurred while designing and constructing the prototype.

To provide a view of the functionality of RePortS, the flow of one vessel through the system is depicted and explained using screen shots of the different web pages of the application. Note that no security measures such as authenticated log in or restricting user access to parts of the system have been implemented in RePortS. Implementing security aspects depends heavily upon the technology used for implementation such as the database system, web server and server operating system. Since the main purpose of the prototype is to show feasibility of the redesigned process, no security measures were included. Two possible methods of providing security for the full, operational system will be mentioned briefly. The first is using the security features of IIS and Microsoft Windows NT Server. When using these features, access to certain parts of the web site is restricted to only those who supply a valid Windows NT user name and password. Additionally, the user name and password are encrypted for transmission. The second method is to use the underlying database system to provide user names and passwords and to use Secure Sockets Layer (SSL) to provide user name and password encryption. Since the CGWEB is an intranet and is only accessible to those behind the firewall, the use of Windows NT user names and passwords could be used for those portions of the web site that are for Coast Guard use only. For the vessel agents, unique user names and passwords could be supplied and SSL could be used for encryption.

The home page of the system is a page that welcomes the user and presents a number of options. To begin the tour of the system, the hyperlink on the home page (Figure 5-4) for entering a vessel arrival is clicked. This sends the user to another page, shown in Figure 5-5, which allows the entry of a vessel name.

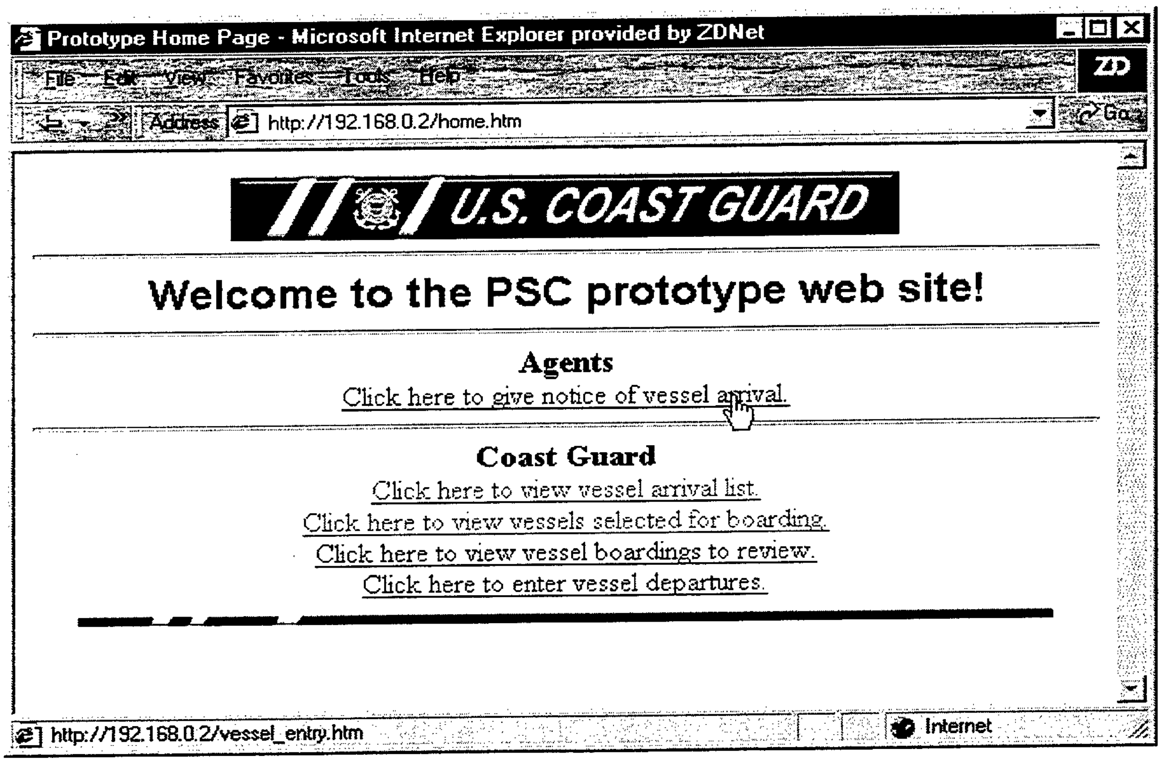


Figure 5-4. Prototype Home Page

When the submit button is clicked, the vessel name is searched for in the database on the server. If there is a match, a page is presented to the user with all matches and additional identifying information on the matching vessels (Figure 5-6).

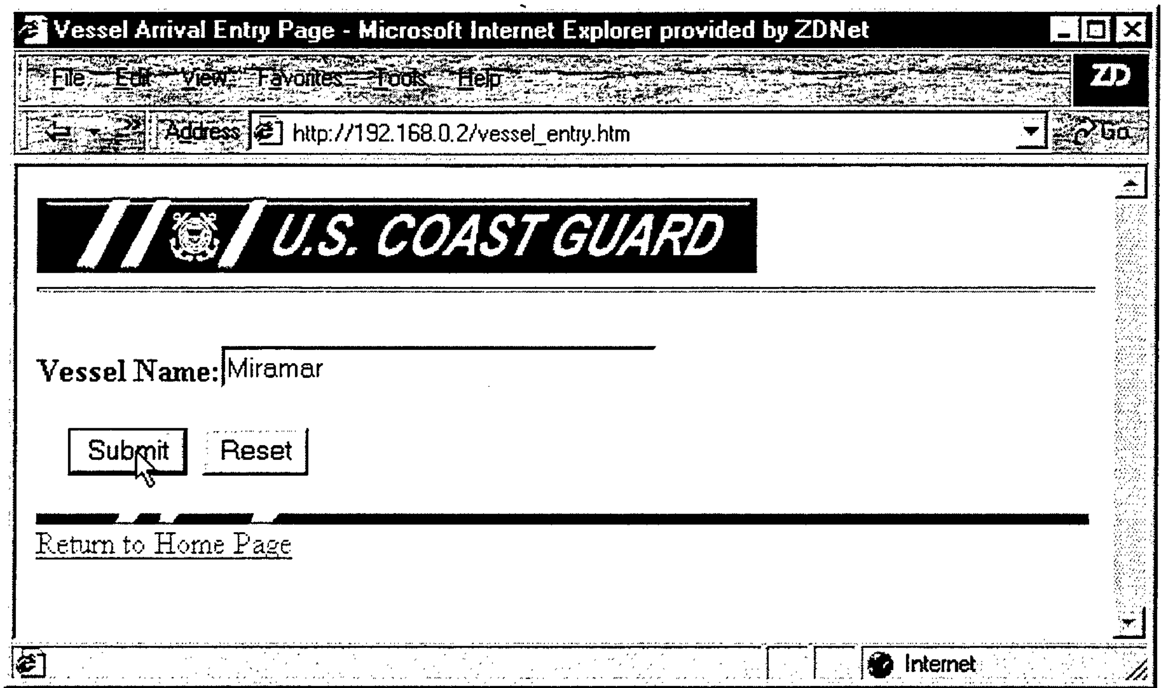


Figure 5-5. Vessel Name Entry Page

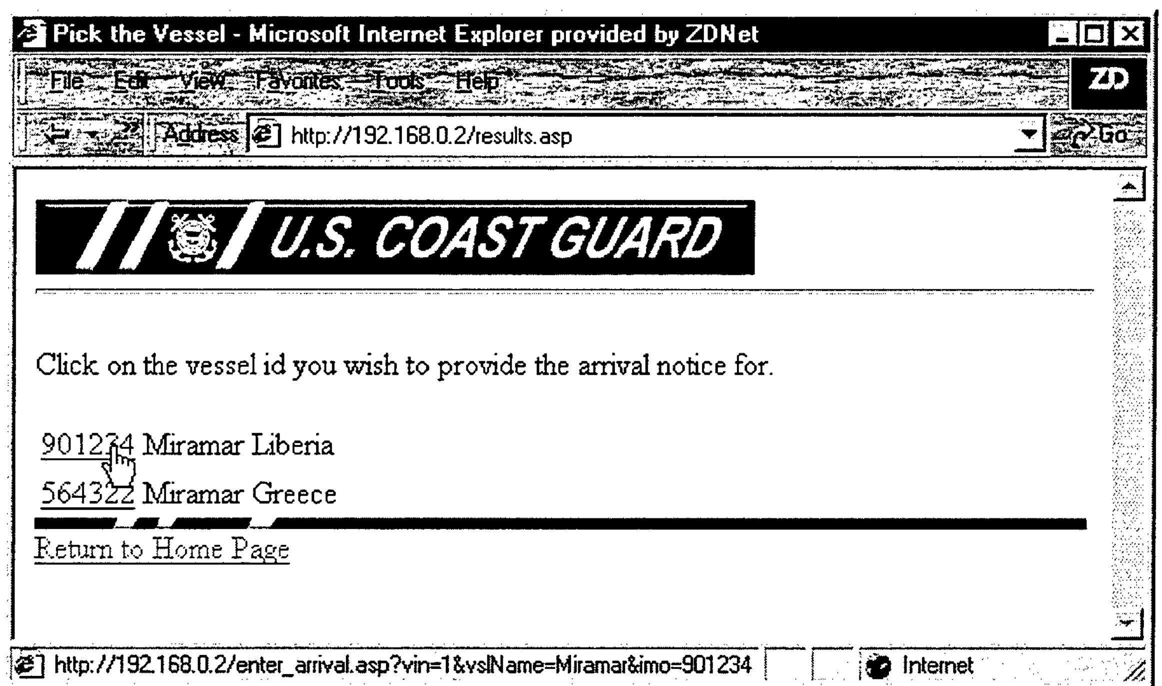


Figure 5-6. Selection of Vessel

Now the user can select the appropriate vessel by clicking on the IMO number of the vessel. This brings up a page that allows the user to enter the information for the arrival of the vessel using text boxes and drop down list boxes. This is shown in Figure 5-7. When the user clicks on the submit button, the information is sent to the server. The server then updates the data in the database, gathers information for the grading of the vessel, grades the vessel, and updates the priority of the vessel. A confirmation is returned to the user of the acceptance of the arrival notice along with a message on the priority and likelihood of a PSC boarding (Figure 5-8).

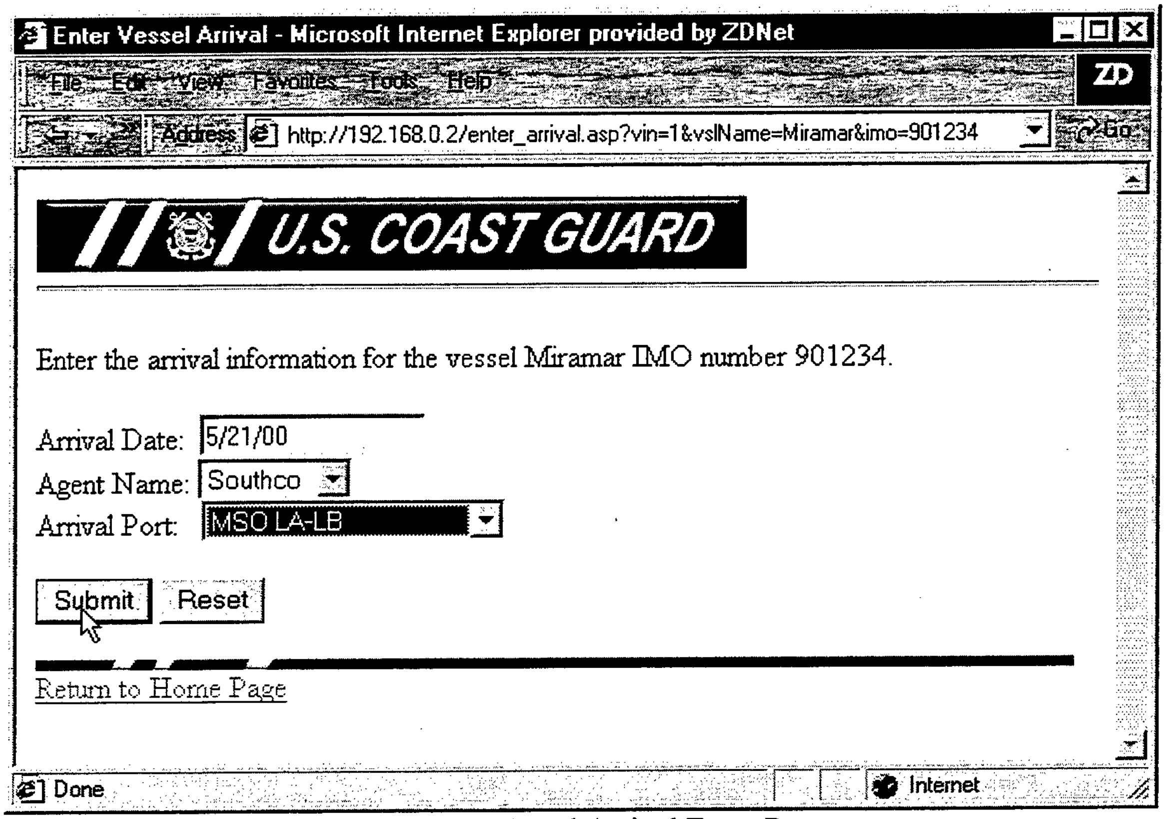


Figure 5-7. Vessel Arrival Entry Page

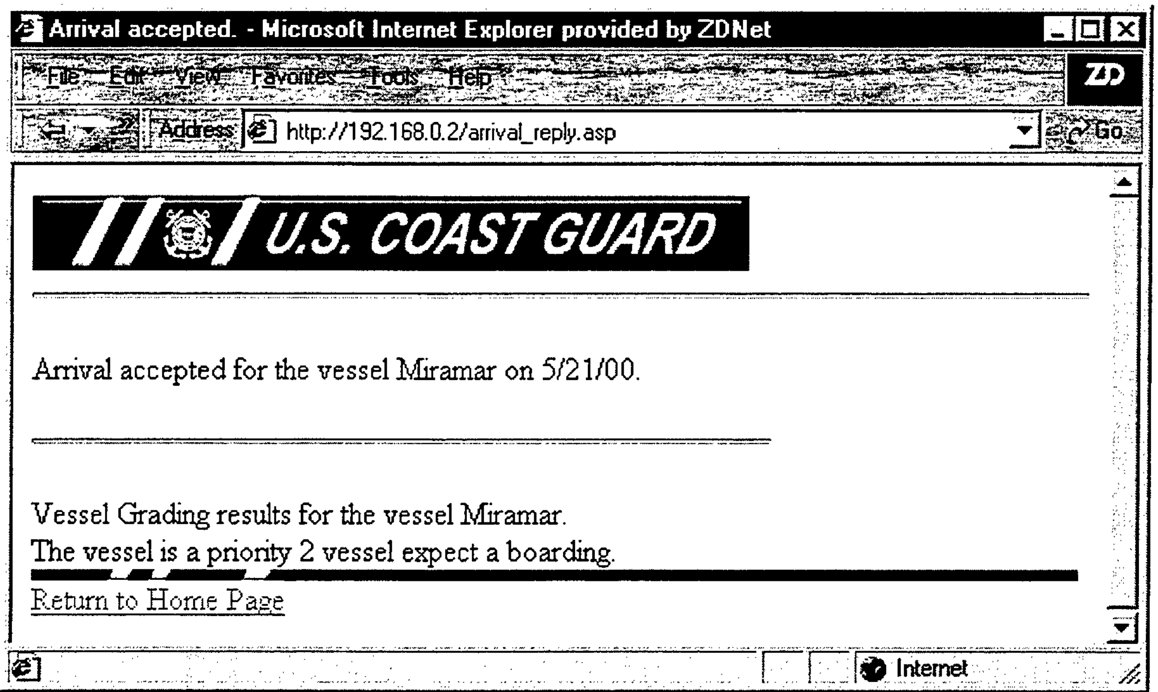


Figure 5-8. Arrival Acceptance and Priority Read Out

With the vessel arrival entered into the system, the PSC supervisor can now click on the link for viewing the vessel arrivals in the port. (See Figure 5-4) This link takes the user to a page where he can enter the port for the list of arrivals he desires. (See Figure 5-9)

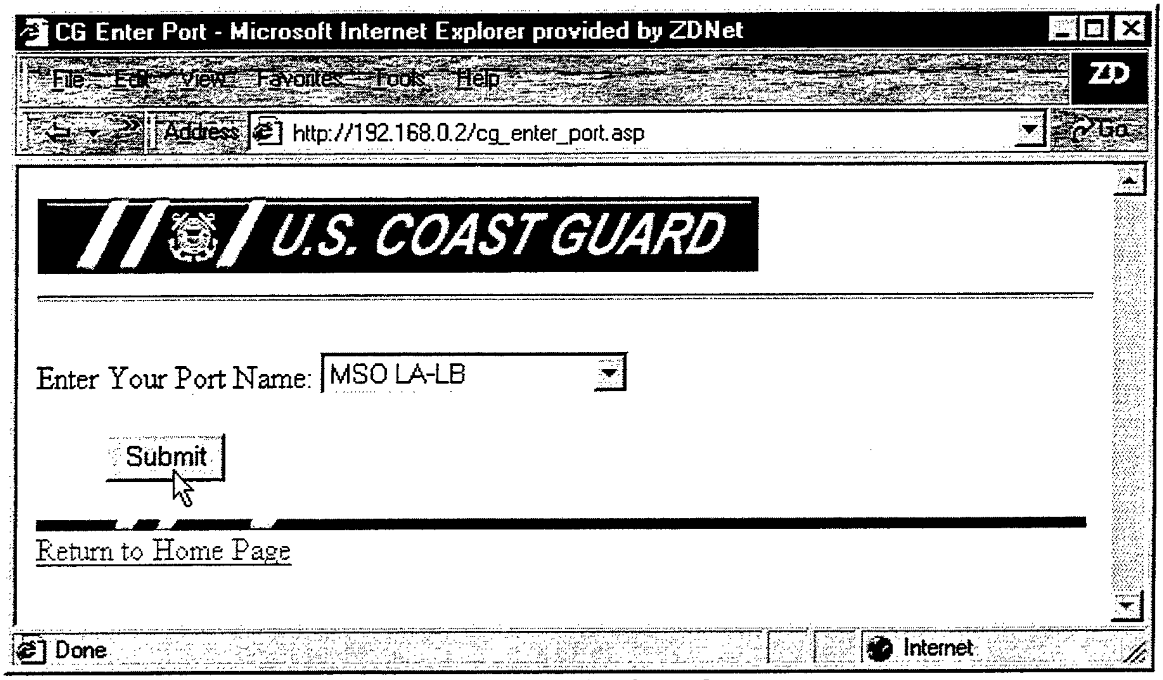


Figure 5-9. Selection of Port

After providing the information, a list of all the vessel arrivals to the port is presented with those scoring the highest number of points in the grading process at the top of the list. This provides the decision support for selecting the vessels to board. The list has check boxes next to each vessel for the user to select which vessels are targeted for boarding that day. (See Figure 5-10) When the user has checked the vessels he wants boarded, he clicks on the submit button, and the selected vessel boarding records are updated. The user is then presented with an acknowledgement of his actions.

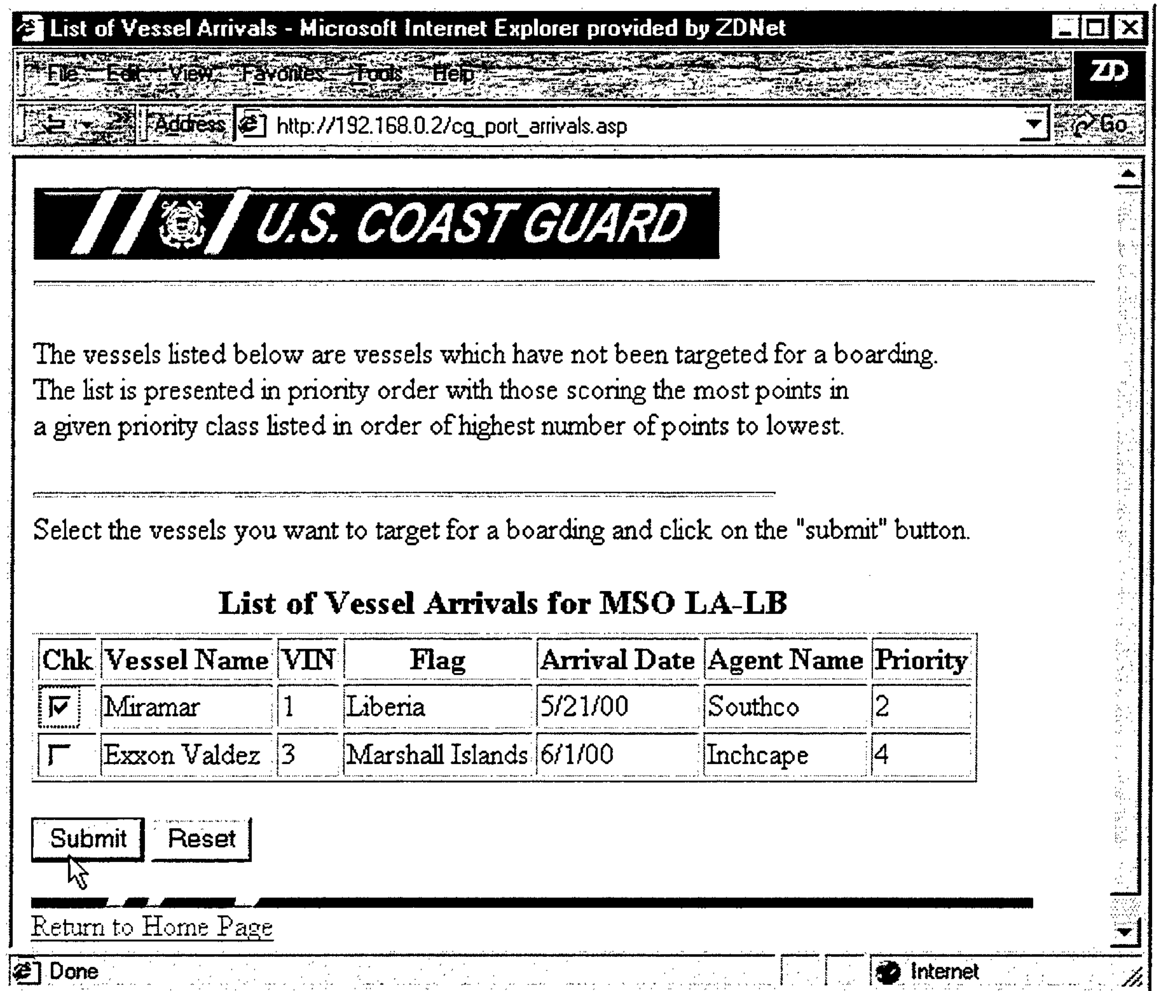


Figure 5-10. List of Vessel Arrivals

The boarding officer assigned to the boarding selects the link to view vessels selected for boarding. This generates a list of vessels selected for boarding in the port. The boarding officer selects the vessel(s) he has been assigned by checking the check box and clicking the submit button. (See Figure 5-11) The boarding and vessel data needed for the boarding are downloaded to the portable database and the boarding officer is ready to perform the boarding.

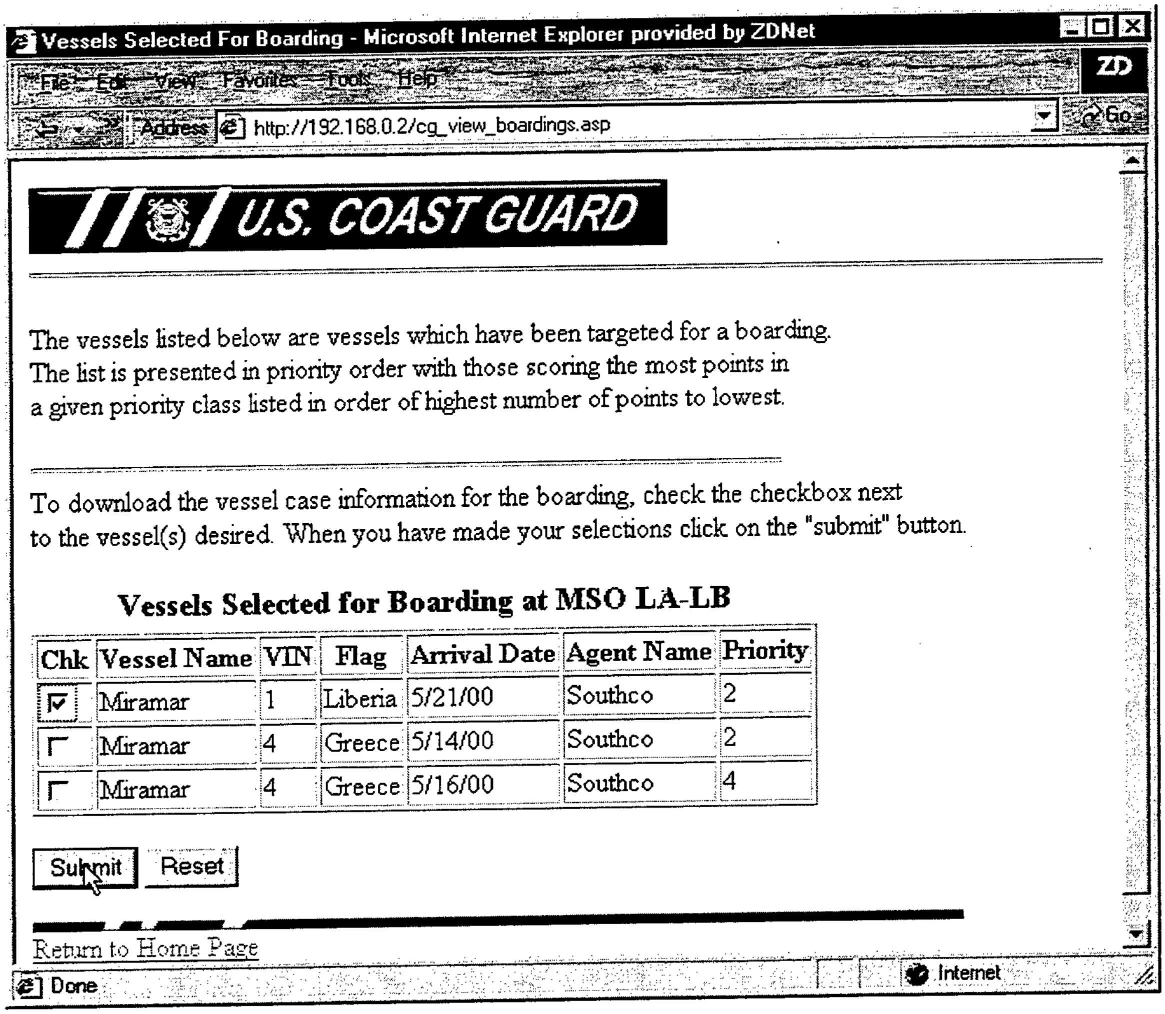


Figure 5-11. Vessels Selected for Boarding

Figure 5-12 shows the menu for the portable application. The options shown on the figure allow the user to access the different forms of the application. Figure 5-13 is a screen shot of a boarding update form that the boarding officer would use while on board the vessel. The next screen shot, Figure 5-14, shows the certificate update page for use on the vessel. When the boarding is complete, a vessel boarding letter is generated from the data entered during the boarding (See Figure 5-15), printed out and left with the Master of the Vessel. When the boarding is completed, the data in the portable database is uploaded to the server, thus updating the vessel and boarding data.

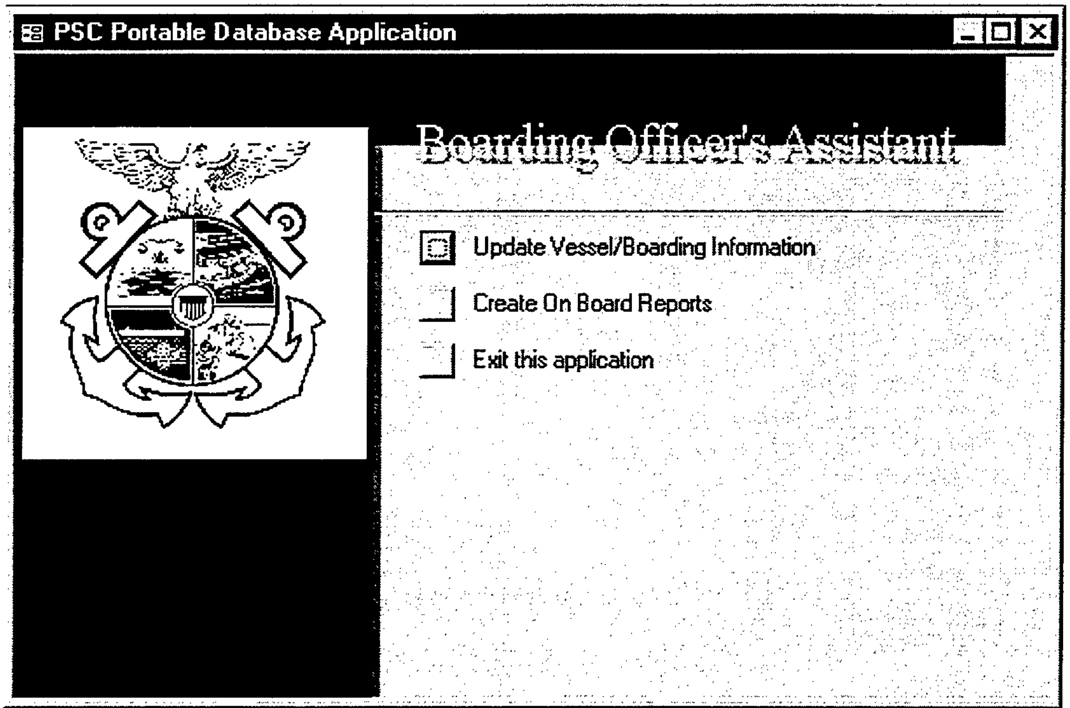


Figure 5-12. Portable Application Menu

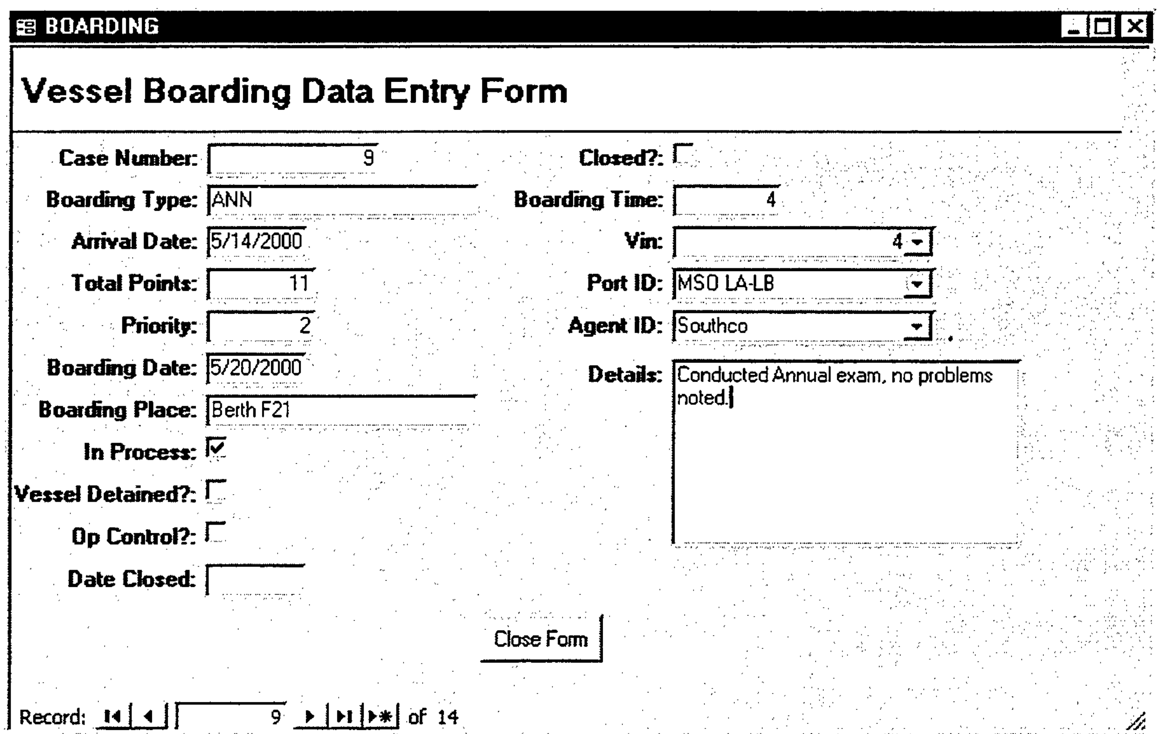
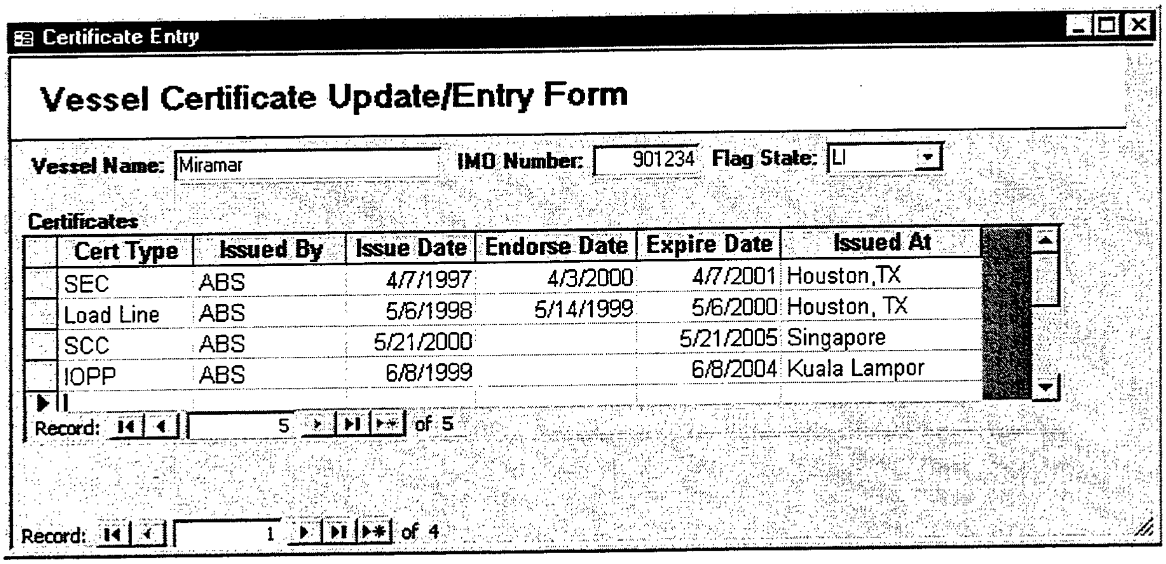


Figure 5-13. Vessel Boarding Entry Form



Certificate Entry

Vessel Certificate Update/Entry Form

Vessel Name: Miramar **IMO Number:** 901234 **Flag State:** LI

Certificates

Cert Type	Issued By	Issue Date	Endorse Date	Expire Date	Issued At
SEC	ABS	4/7/1997	4/3/2000	4/7/2001	Houston,TX
Load Line	ABS	5/6/1998	5/14/1999	5/6/2000	Houston, TX
SCC	ABS	5/21/2000		5/21/2005	Singapore
IOPP	ABS	6/8/1999		6/8/2004	Kuala Lampor

Record: 5 of 5

Record: 1 of 4

Figure 5-14. Vessel Certificate Update Form

USCG Port State Control
Vessel Examination Letter

Master, M/V Miramar IMO Number 564322,

On 5/20/2000 your vessel was examined by officers from MSO LA-LB at Berth F21 .

The results of the examination are described below:

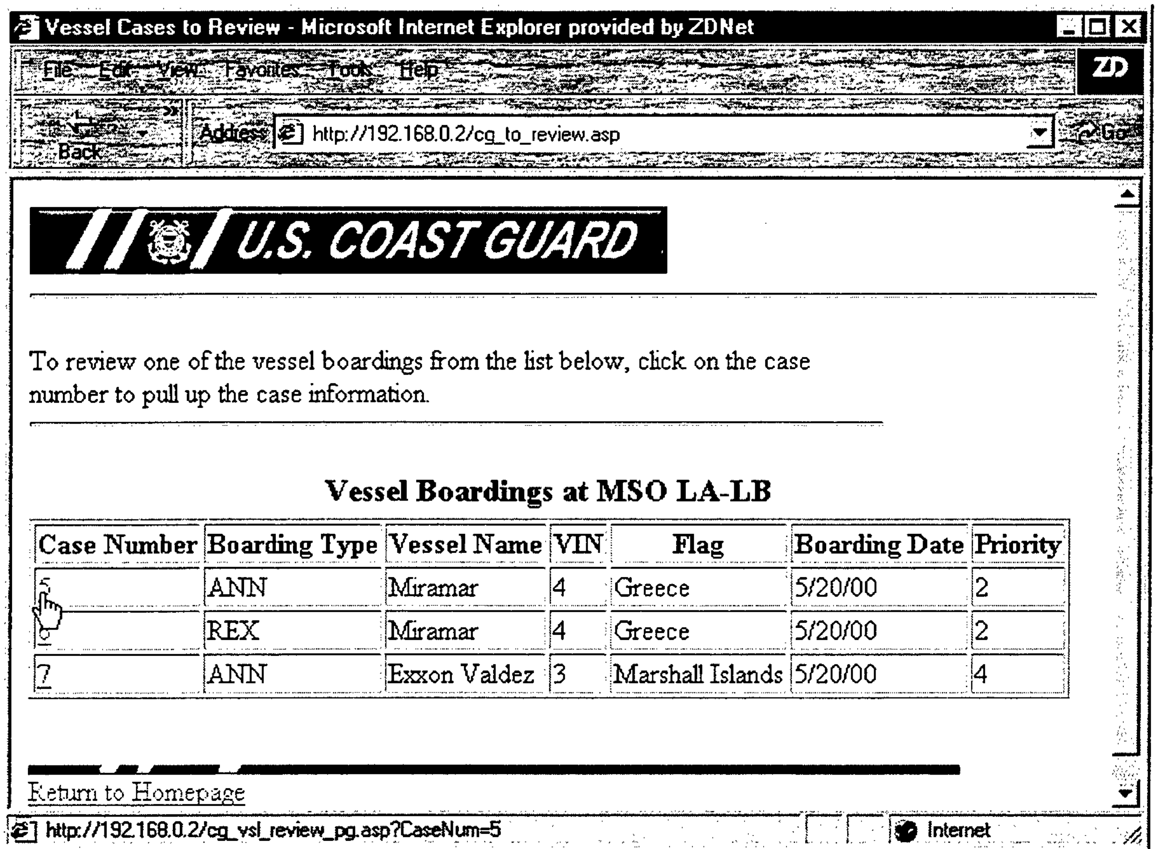
No Discrepacies found.

Please keep a copy of this letter on board and provide it to the next U.S. Coast Guard personnel who board your vessel. A record of this boarding has been made into the U.S. Coast Guard Marine Safety Network and is noted as case number: 9

Boarding Officer Tim Kunckle BM3 USCG

Figure 5-15. Vessel Examination Letter

At this stage, the supervisor can access the list of vessels that have been boarded and know that the information has been synchronized with the server database. This provides a list of vessels from which the supervisor can select to review the boarding case and vessel information updates (See Figure 5-16). Once the supervisor has reviewed and approved the case, the validate box is checked, and the boarding record is updated with the information (See Figure 5-17). The supervisor gets an acknowledgement of the validation and can view additional vessel boarding cases.



Case Number	Boarding Type	Vessel Name	VIN	Flag	Boarding Date	Priority
5	ANN	Miramar	4	Greece	5/20/00	2
6	REX	Miramar	4	Greece	5/20/00	2
7	ANN	Exxon Valdez	3	Marshall Islands	5/20/00	4

Figure 5-16. List of Vessels to Review

The final activity in the PSC process is the logging of vessels that have departed the port and were not boarded, Figure 5-18 shows this list. The list allows the user to select the vessels that have departed the port and to enter the date of departure. When the submit button is clicked, the selected records are updated in the database along with an acknowledgement of the updates.

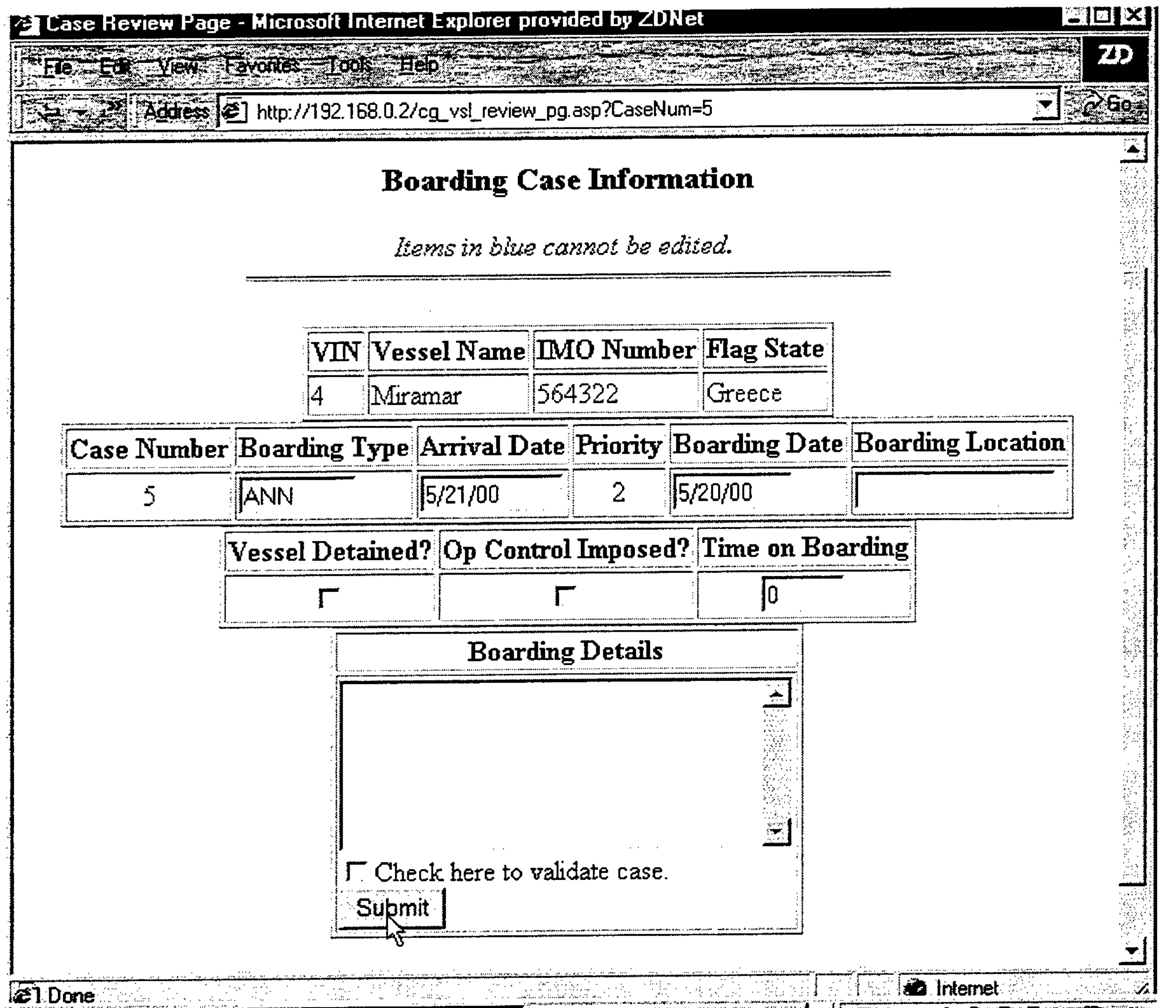


Figure 5-17. Vessel Boarding Review Page

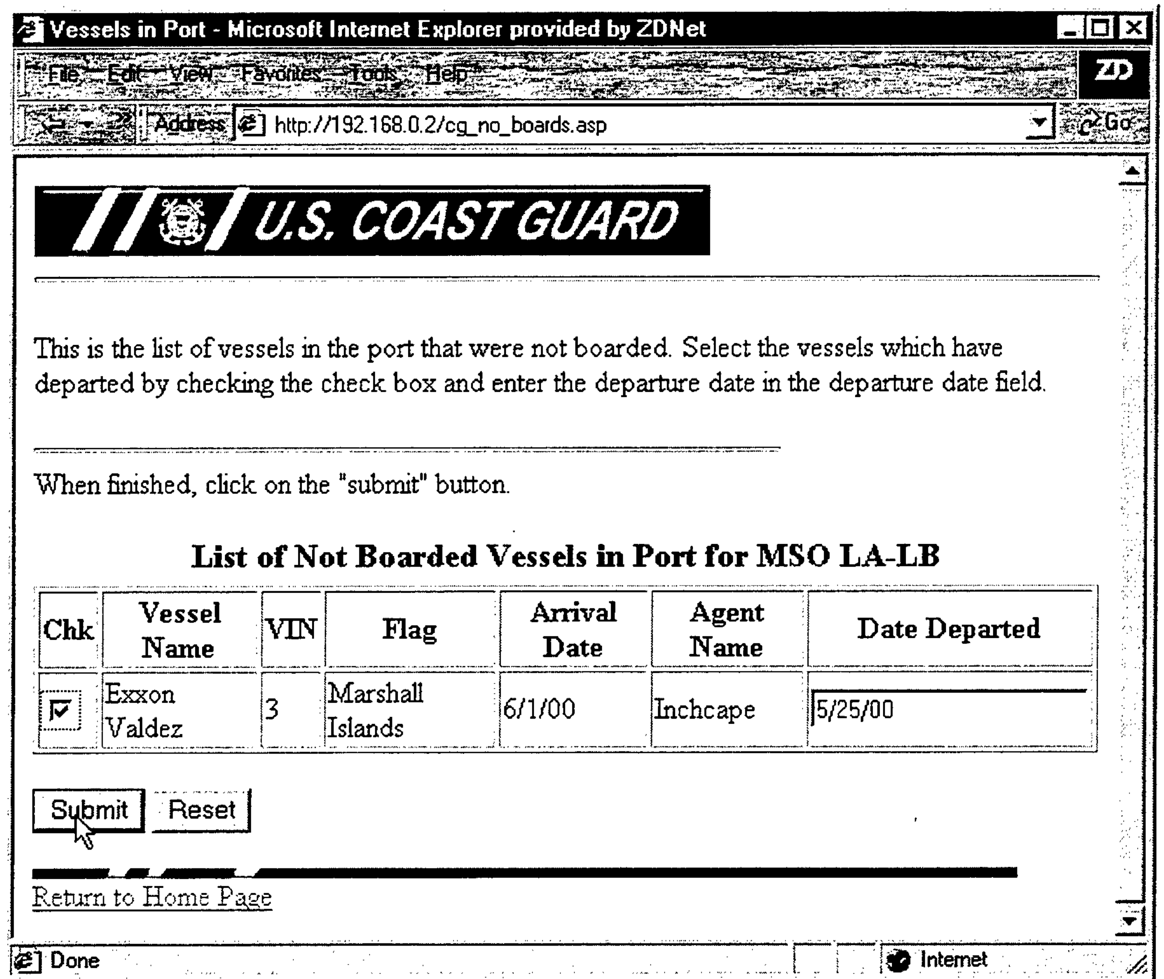


Figure 5-18. List of Vessels Not Boarded

D. SUMMARY

This Chapter has presented the development of a prototype to implement the redesigned process discussed in Chapter IV. The RAD methodology was used in the development process. Requirements and design were presented, data and process models were created, and the construction of the prototype was explained. Finally, RePortS was demonstrated by providing a use case walk through of the system using screen shots to explain the process flow for a particular vessel. The prototype has demonstrated that the implementation of the redesigned PSC process is feasible, and shows how the redesigned process would look in implementation form.

THIS PAGE INTENTIONALLY LEFT BLANK

VI. CONCLUSIONS AND RECOMMENDATIONS

The main purpose of this thesis is to analyze and perform a business process redesign of the U. S. Coast Guard Port State Control process. The focus is to provide innovative solutions that dramatically improve the cycle time and data accuracy of the process.

A. CONCLUSIONS

The current process consists of sixteen activities ranging from the vessel agent calling in a vessel arrival to the final step of logging the departure of the vessel. The goals of the process are timeliness and accuracy in the identification of vessels requiring boarding, in the gathering of data on the current state of a vessel being boarded, and in the documentation of vessel boardings. Process cycle time and data accuracy are the critical performance measures for the process.

Of the three major approaches to BPR, process reengineering, process redesign and continuous process improvement, process redesign was selected as the most suitable methodology. The steps to accomplish the redesign are specified as:

1. Process Identification
2. Process Modeling
3. Configuration Measurement
4. Pathology Diagnosis
5. Matching of Transformations
6. Generation of Redesigns
7. Testing of Alternatives
8. Selection
9. Implementation

The overall PSC process was decomposed into two separate processes, the targeting process and the vessel boarding process. Each of the processes was analyzed by the KOPeR system to identify potential pathologies with respect to the process measures of size, length, handoffs, feedback loops, IT support, IT communication and IT automation. The pathologies identified for both processes were:

1. Sequential Process
2. Process Friction
3. Inadequate IT Support
4. Inadequate IT Communications
5. Inadequate IT Automation

The recommendations for the targeting process redesign were to:

1. Delinearize by automating the call in and grading activities, which would allow them to operate in parallel.
2. Create a case manager by automating the call in and grading process, and provide an automated decision support mechanism.
3. Provide additional IT support by leveraging the CGSWIII and MSN contracts.
4. Provide IT communication by using MSN to keep records and provide the lists traditionally kept on paper.
5. Provide IT automation by having a computer perform the grading and logging activities of the process.

The recommendations for the vessel boarding process were to:

1. Provide additional IT support by using the CGSWIII and MSN to support the process and use a portable database application to support operations on board a vessel.
2. Provide IT communication by using MSN to keep records and use a portable database application with a synchronizing method to pass boarding information.
3. Provide IT automation by keeping the official record of vessel boardings electronically.

Simulation models were constructed of the current and proposed redesigned processes with simulation runs conducted to gather data for the critical performance measures. Each redesigned process incorporated simulation runs with three different scenarios; *most likely*, *optimistic* and *pessimistic*. This provided a balanced look at the possible gains provided by the redesign effort. The redesigned processes were then compared against the current processes to identify areas of improvement in the critical performance measures. The results of the simulation models showed a potential savings of between 5821.75 and 2308.6 hours per year for the targeting process and a potential savings of between 4777.9 and 829.8 hours per year for the vessel boarding process. The simulation results provide evidence of convincing time savings resulting from the redesign.

A prototype of the system required to implement the redesigned PSC process was developed. Tools were identified and expectations set for the prototype. The RAD method of systems development was selected as the development methodology and discussed. Requirements were identified from Chapter IV and documented using a SOM, database schema, and DFD. The prototype was constructed using a database, ASP, VBScript, and a portable Microsoft Access database application. The prototype application was demonstrated via the execution of a use case study to show the feasibility of implementing a system redesign as proposed.

This thesis has conducted an analysis and redesign of the PSC process using BPR methods, discrete event simulation and a prototype implementation of the redesign. The simulation model indicates that implementing the modifications of the process design would result in significant manpower savings, ranging from 2 to 4 person years. The prototype establishes convincing proof of concept indicating the redesign is eminently feasible.

B. RECOMMENDATIONS

Based upon the results above, the following recommendations are made:

1. The Coast Guard should implement the proposed redesign presented in this thesis. The course of implementation should begin by providing the targeting process functionality with MSN. The vessel boarding process should be initially started as a pilot program in several of the larger ports to validate the simulation results and identify costs regarding implementation.

2. During the course of this research, the Coast Guard has independently implemented some of the ideas presented in this thesis with the MSN, for example, the automated grading and targeting of vessels. With a team of developers already working on MSN, additional features identified in this thesis, such as the portable database application and vessel agent provided web based vessel arrival information, should be explored for subsequent implementation in MSN.
3. The premise of this thesis can be extended to other marine safety processes that will be customers of the MSN, such as the casualty investigation process, the domestic vessel boarding process, and the oil spill investigation process. These extensions could be the topic for a "follow on" thesis, or the Coast Guard could use similar methods to those implemented in this thesis to perform a BPR on the processes themselves.

C. FUTURE RESEARCH

The following are areas identified for future research considerations:

1. The prototype was unable to be field tested to provide actual numbers for comparison against the simulation model. Another area of research is be the complete production of a prototype, building upon the work already done with MSN in conjunction with this thesis, for use in performing a field test of the system. This would provide additional insight into the validity of the simulation models and their extensibility to other processes.
2. Finally, an additional area for further research is the development of a more robust decision support/expert system in determining the highest risk vessels entering a port. The PSC program has been operating for five full years and there is a large amount of data in MSIS that could be put into a data warehouse and analyzed using data mining techniques. With the information gleaned from this data, specific risk profiles could be identified to assist with the targeting process.

APPENDIX A. PSC TARGETING MATRIX

This is a copy of the Port State Control Targeting Matrix from the Coast Guard Marine Safety Manual, Volume II.

OWNER	FLAG	CLASS	HISTORY	SHIP TYPE
5 Points	**7 Points**	**Priority 1**	**5 Points Each**	**1 Point**
Listed Owner or Operator	Listed Flag State	≥10 arrivals with detention ratio more than 4 times the average OR <10 arrivals and involved with at least one detention in the previous 3 years. **5 Points** ≥10 arrivals with a detention ratio between 3 & 4 times the average. **3 Points** ≥10 arrivals with a detention ratio between 2 & 3 times the average. **1 Point** ≥10 arrivals with a detention ratio between the average and twice the average. **0 Points** ≥10 arrivals with a detention ratio below the average **OR** <10 arrivals with no detentions in the previous 3 years.	Detention within the previous 12 months. **1 Point Each** Other operational control within the previous 12 months **1 Point Each** Casualty within the previous 12 months. **1 Point Each** Violation within the previous 12 months. **1 Point Each** Not boarded within the previous 6 months.	Oil or chemical Tanker **1 Point** Gas Carrier **2 Points** Bulk Freighter over 10 years old. **1 Point** Passenger Ship **2 Points** Carrying low value commodities in bulk.

THIS PAGE INTENTIONALLY LEFT BLANK

APPENDIX B. DATA DEFINITION

This appendix contains the data definition for the database schema described in Chapter V.

Table	Attributes	Type (size)	Key/Foreign Key
VESSEL	Vin	AutoNumber	Key
	IMONumber	Number	
	VslName	Text(25)	
	BldDate	Date/Time	
	VslType	Text(3)	
	Length	Number	
	Breadth	Number	
	Depth	Number	
	Propulsion	Text(15)	
	SpecialNote	Text(240)	
	ClassID_FK	Number	FK
	FlagID_FK	Number	FK
BOARDING	CaseNum	AutoNumber	Key
	Type	Text(5)	
	Tpoints	Number	
	Priority	Number	
	ArrivalDate	Date/Time	
	BdngDate	Date/Time	
	BdngPlace	Text(35)	
	InProc	Yes/No	
	Detained	Yes/No	
	OpControl	Yes/No	
	DateClosed	Date/Time	
	Closed	Yes/No	
	Details	Text(254)	
	BoardingTime	Number	
	Vin_FK	Number	FK
	PortID_FK	Number	FK
	PartyID_FK	Number	FK

Table	Attributes	Type (size)	Key/Foreign Key
CLASS	ClassID	AutoNumber	Key
	ClassCode	Text(6)	
	ClassName	Text(35)	
	Street	Text(20)	
	City	Text(20)	
	State	Text(4)	
	Country	Text(3)	
	Zip	Number	
	PhoneNo	Text(15)	
	TPoints	Number	
INTPARTY	PartyID	AutoNumber	Key
	PartyName	Text(25)	
	PartyRole	Text(10)	
	Street	Text(20)	
	City	Text(20)	
	State	Text(4)	
	Country	Text(3)	
	Zip	Number	
	PhoneNo	Text(15)	
	TPoints	Number	
DEF	DefID	Number	Key
	CaseNum_FK	Number	Key/FK
	DefType	Text(25)	
	DefCode	Text(5)	
	FixBy	Date/Time	
	Problem	Text(50)	
	DateClosed	Date/Time	
	Closed	Yes/No	
FLAGSTATE	FlagID	AutoNumber	Key
	FlagCode	Text(3)	
	CountryName	Text(15)	
	Street	Text(20)	
	City	Text(20)	
	State	Text(4)	
	Zip	Number	
	PhoneNo	Text(15)	
	TPoints	Number	
VESSEL-INTPARTY	Vin_FK	Number	Key/FK
	PartyID_FK	Number	Key/FK

Table	Attributes	Type (size)	Key/Foreign Key
CERTIFICATE	CertType	Text(7)	Key
	Vin_FK	Number	Key/FK
	IssuedBy	Text(15)	
	IssueDate	Date/Time	
	EndorseDate	Date/Time	
	ExpireDate	Date/Time	
	IssuedAt	Text(15)	
BOARDING OFFICER	BOID	AutoNumber	Key
	BOName	Text(25)	
	Rank	Text(7)	
	Qual	Text(25)	
	QualDate	Date/Time	
	PortID_FK	Number	FK
PORT	PortID	AutoNumber	Key
	UnitName	Text(25)	
	Street	Text(20)	
	City	Text(20)	
	State	Text(4)	
	Zip	Number	
	PhoneNo	Text(15)	
	Email	Text(35)	
BOARDING-BOARDING OFFICER	CaseNum_FK	Number	Key/FK
	BOID_FK	Number	Key/FK
DEFRES	DefID_FK	Number	Key/FK
	CaseNum_FK	Number	Key/FK
	Resolution	Text(100)	
	CaseNumRes_FK	Number	FK

THIS PAGE INTENTIONALLY LEFT BLANK

APPENDIX C. REPORTS WEB PAGE CODE

This appendix provides the code behind the web pages created for the implementation of RePortS.

A. HOME.HTM

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.0 Transitional//EN">
<html>
<head>
	<title>Prototype Home Page</title>
</head>
<body>
<center><img src="images/medbar.jpg" width="368" height="39" alt=""
border="0"><br>
<hr size="3" width="100%">
<strong><font face="arial" size="5">Welcome to the PSC prototype web
site!</font></strong>
<hr size="3" width="100%">
<center><b><big>Agents</big></b></center>
<a href="vessel_entry.htm">Click here to give notice of vessel arrival.</a><br>
<hr size="3" width="100%">
<center><b><big>Coast Guard</big></b></center>
<a href="cg_enter_port.asp">Click here to view vessel arrival list.</a><br>
<a href="cg_enter_port1.asp">Click here to view vessels selected for boarding.</a><br>
<a href="cg_enter_port2.asp">Click here to view vessel boardings to review.</a><br>
<a href="cg_enter_port3.asp">Click here to enter vessel departures.</a>
<img src="images/rc_rod2.gif" width="538" height="6" alt="" border="0">
</body>
</html>
```

B. CG_ENTER_PORT.ASP

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.0 Transitional//EN">
<html>
<head>
	<title>CG Enter Port</title>
</head>
<img src="images/medbar.jpg" width="368" height="39" border="0">
<body>
<hr size="3" width="100%"><br>
<form action="cg_port_arrivals.asp" method="POST" name="cgenterport">
Enter Your Port Name: <!--#include file="portdown.asp"-->
```

```
<br><br>
        
<input type="submit" value="Submit">
</form>
</body>
<img src="images/rc_rod2.gif" width="538" height="6" border="0"><br>
<a href="home.htm">Return to Home Page</a>
</html>
```

C. PORTDOWN.ASP

```
<%
SQLPORT="SELECT PortID, UnitName FROM PORT"
set connport = server.createobject("ADODB.Connection")
connport.open "test2"
set ports=connport.execute(SQLPORT)
%>
<select name="PortID">
                <% do while not ports.eof %>
                  <Option value = "<%= ports(0) %>"> <%= ports(1) %></Option>
                <%ports.movenext
                loop%>
</select>
<% connport.close %>
```

D. CG_PORT_ARRIVALS.ASP

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.0 Transitional//EN">
<html>
<head>
        <title>List of Vessel Arrivals</title>
</head>
<img src="images/medbar.jpg" width="368" height="39" border="0">
<body>
<hr size="3" width="100%"><br>
<%
port = Request.Form("PortID")
'set up SQL to get vessel recordset
SQL= "SELECT CaseNum, ArrivalDate, Priority, Vin_FK, PartyID_FK FROM
BOARDING WHERE PortID_FK = " & port & " AND"
SQL= SQL & " Type = 'NOBD' AND Closed = False ORDER BY TPoints DESC;"
'get port name for completeness
PSQL = "SELECT UnitName FROM PORT WHERE PortID = " & port & ";"
'set up connection to the database
```

```
set conn1 = server.createobject("ADODB.Connection")
conn1.open "test2"
set rsBoarding = conn1.execute(SQL)
set rsPort = conn1.execute(PSQL)
%>
The vessels listed below are vessels which have not been targeted for a boarding.<br>
The list is presented in priority order with those scoring the most points in <br>
a given priority class listed in order of highest number of points to lowest.<br><br>
<hr size="3" width="70%" align="left">
Select the vessels you want to target for a boarding and click on the "submit" button.
<form action="cg_boarding_pick_confirmation.asp" method="POST" name="list">
<table border>
<Caption><b><big>List of Vessel Arrivals for <%=rsPort(0)%></big></b></Caption>
<THEAD>
        <TR>
         <TH>Chk</TH>
              <TH>Vessel Name</TH>
              <TH>VIN</TH>
              <TH>Flag</TH>
              <TH>Arrival Date</TH>
              <TH>Agent Name</TH>
              <TH>Priority</TH>
        </TR>
</THEAD>
<TBODY>
<%
do while not rsBoarding.eof
VSQL= "SELECT VslName, FlagID_FK FROM VESSEL WHERE Vin = " &
rsBoarding(3) & ";"
set rsVessel = conn1.execute(VSQL)
FSQL= "SELECT CountryName FROM FLAGSTATE WHERE FlagID = " &
rsVessel(1) & ";"
set rsFlag = conn1.execute(FSQL)
ASQL= "SELECT PartyName FROM INTPARTY WHERE PartyID = " &
rsBoarding(4) & ";"
set rsAgent = conn1.execute(ASQL)
%>
<tr>
    <td><input type="checkbox" name="box" value="<%=rsBoarding(0)%>"></td>
    <td><%=rsVessel(0)%></td>
        <td><%=rsBoarding(3)%></td>
        <td><%=rsFlag(0)%></td>
        <td><%=rsBoarding(1)%></td>
        <td><%=rsAgent(0)%></td>
```

```
	<td><%=rsBoarding(2)%></td>
</tr>
<%rsBoarding.MoveNext
loop%>
</TBODY>
</table><br>
<input type="submit" value="Submit"> <input type="reset">
</form>
</body>
<img src="images/rc_rod2.gif" width="538" height="6" border="0"><br>
<a href="home.htm">Return to Home Page</a>
</html>
```

E. CG_BOARDING_PICK_CONFIRMATION.ASP

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.0 Transitional//EN">
<html>
<head>
	<title>Confirmation of Vessels</title>
</head>
<img src="images/medbar.jpg" width="368" height="39" border="0">
<body>
<hr size="3" width="100%"><br>
<%
'get today's date
dTDate = Date
'set up connection to the database
set conn1 = server.createobject("ADODB.Connection")
conn1.open "test2"
'loop through the checkboxes passed from the last page
For Each item In Request.Form("box")
 'update record to indicate vessel is selected for a boarding
 USQL = "UPDATE BOARDING SET Type = 'BD', BdngDate = '" & dTDate & "'
WHERE CaseNum = " & item & ";"
 conn1.execute(USQL)
Next
%>
Vessel(s) selected for boardings updated.
<br>
</body>
<img src="images/rc_rod2.gif" width="538" height="6" border="0"><br>
<a href="home.htm">Return to Home Page</a>
</html>
```

F. CG_VIEW_BOARDINGS.ASP

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.0 Transitional//EN">
<html>
<head>
        <title>Vessels Selected For Boarding</title>
</head>
<img src="images/medbar.jpg" width="368" height="39" border="0">
<body>
<hr size="3" width="100%"><br>
<%
port = Request.Form("PortID")
'set up SQL to get vessel recordset
SQL= "SELECT CaseNum, ArrivalDate, Priority, Vin_FK, PartyID_FK FROM
BOARDING WHERE PortID_FK = " & port & " AND"
SQL= SQL & " Type = 'BD' AND Closed = False ORDER BY TPoints DESC;"
'get port name for completeness
PSQL = "SELECT UnitName FROM PORT WHERE PortID = " & port & ";"
'set up connection to the database
set conn1 = server.createobject("ADODB.Connection")
conn1.open "test2"
set rsBoarding = conn1.execute(SQL)
set rsPort = conn1.execute(PSQL)
%>
The vessels listed below are vessels which have been targeted for a boarding.<br>
The list is presented in priority order with those scoring the most points in <br>
a given priority class listed in order of highest number of points to lowest.<br><br>
<hr size="3" width="70%" align="left">
To download the vessel case information for the boarding, check the checkbox next<br>
to the vessel(s) desired. When you have made your selections click on the "submit"
button.<br>
<form action="cg_download_db.asp" method="POST" name="list">
<table border>
<Caption><b><big>Vessels Selected for Boarding at
<%=rsPort(0)%></big></b></Caption>
<THEAD>
          <TR>
           <TH>Chk</TH>
                <TH>Vessel Name</TH>
                <TH>VIN</TH>
                <TH>Flag</TH>
                <TH>Arrival Date</TH>
                <TH>Agent Name</TH>
                <TH>Priority</TH>
```

```
        </TR>
</THEAD>
<TBODY>
<%
do while not rsBoarding.eof
VSQL= "SELECT VslName, FlagID_FK FROM VESSEL WHERE Vin = " &
rsBoarding(3) & ";"
set rsVessel = conn1.execute(VSQL)
FSQL= "SELECT CountryName FROM FLAGSTATE WHERE FlagID = " &
rsVessel(1) & ";"
set rsFlag = conn1.execute(FSQL)
ASQL= "SELECT PartyName FROM INTPARTY WHERE PartyID = " &
rsBoarding(4) & ";"
set rsAgent = conn1.execute(ASQL)
%>
<tr>
    <td><input type="checkbox" name="box" value="<%=rsBoarding(0)%>"></td>
    <td><%=rsVessel(0)%></td>
        <td><%=rsBoarding(3)%></td>
        <td><%=rsFlag(0)%></td>
        <td><%=rsBoarding(1)%></td>
        <td><%=rsAgent(0)%></td>
        <td><%=rsBoarding(2)%></td>
</tr>
<%rsBoarding.MoveNext
loop%>
</TBODY>
</table><br>
<input type="submit" value="Submit"> <input type="reset">
</form>
</body>
<img src="images/rc_rod2.gif" width="538" height="6" border="0"><br>
<a href="home.htm">Return to Home Page</a>
</html>
```

G. AGENTDOWN.ASP

```
<%
SQLAGENT="SELECT PartyID, PartyName FROM INTPARTY WHERE PartyRole =
'agent'"
set connagent = server.createobject("ADODB.Connection")
connagent.open "test2"
set agents=connagent.execute(SQLAGENT)
%>
```

```
<select name="AgentID">
                <% do while not agents.eof %>
                  <Option value = "<%= agents(0) %>"> <%= agents(1) %></Option>
                <%agents.movenext
                loop%>
</select>
<% connagent.close %>
```

H. ENTER_ARRIVAL.ASP

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.0 Transitional//EN">
<html>
<head>
        <title>Enter Vessel Arrival</title>
</head>
<body>
<img src="images/medbar.jpg" width="368" height="39" border="0"><br>
<hr size="3" width="100%"><br>
<%
vin = Request.Querystring("vin")
vslName=Request.Querystring("vslName")
imo=Request.Querystring("imo")%>
Enter the arrival information for the vessel <%=vslName%> IMO number
<%=imo%>.<br>
<form action="arrival_reply.asp" method="POST" name="arrentry">
Arrival Date:  <input name="adate" type="text" align="TOP" size="15"><br>
Agent Name: <!--#include file="agentdown.asp"--><br>
Arrival Port:  <!--#include file="portdown.asp"--><br><br>
<input name="vin" type="hidden" value="<%=vin%>">
<input name="vslName" type="hidden" value="<%=vslName%>">
<input type="submit" value="Submit"> <input type="reset">
</form>
</body>
<img src="images/rc_rod2.gif" width="538" height="6" border="0"><br>
<a href="home.htm">Return to Home Page</a>
</html>
```

I. ARRIVAL_REPLY.ASP

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.0 Transitional//EN">
<!-- #Include file="ADOVBS.INC" -->
<% Language = VBScript %>
<html>
<head>
```

```
        <title>Arrival accepted.</title>
</head>
<img src="images/medbar.jpg" width="368" height="39" border="0">
<body>
<hr size="3" width="100%"><br>
<!-- ADO Connection Object used to create recordset-->
<%
'Create and Open Connection Object
Set OBJdbConnection = Server.CreateObject("ADODB.Connection")
OBJdbConnection.Open "test2"
'Create and Open Recordset Object
Set RsBoardingList = Server.CreateObject("ADODB.Recordset")
RsBoardingList.ActiveConnection = OBJdbConnection
RsBoardingList.CursorType = adOpenKeyset
RsBoardingList.LockType = adLockOptimistic
RsBoardingList.Source = "BOARDING"
RsBoardingList.Open
vin=Request.form("vin")
agent=Request.form("AgentID")
adate=CDate(Request.form("adate"))
port=Request.form("PortID")
vslName = Request.form("vslName")
                RsBoardingList.AddNew
                RsBoardingList("ArrivalDate") = adate
                RsBoardingList("Vin_FK") = vin
                RsBoardingList("PortID_FK") = port
                RsBoardingList("PartyID_FK") = agent
                RsBoardingList("Type") = "NOBD"
                RsBoardingList.Update
                RsBoardingList.MoveLast
%>
Arrival accepted for the vessel <%=vslName%> on <%=adate%>.<br><br>
<%
'declare targeting variables
dim iOwOp, iFlag, iClass, iHistory, iShipType, iTotal, iPriority
'setup variables for use in targeting
'vin = Request.Querystring("vin")
dToday = Date
dOneYear = DateAdd("yyyy", -1, dToday) 'get date for a year ago
dTenYears = DateAdd("yyyy", -10, dToday) 'get a date for ten years ago
dSixMonths = DateAdd("m", -6, dToday) 'get a date six months ago
'setup query to get vessel info
VSQL = "SELECT VslName, VslType, BldDate, FlagID_FK, ClassID_FK FROM "
VSQL = VSQL & "VESSEL WHERE Vin = " & vin & ";"
```

```
'setup query to get owner and operator for vessel
OSQL = "SELECT PartyID_FK FROM [VESSEL-INTPARTY] WHERE Vin_FK = " &
vin & ";"
'setup query to get boardings for the vessel
BSQL = "SELECT CaseNum, Type, BdngDate, VslDetained, OpControl FROM
BOARDING "
BSQL = BSQL & "WHERE Vin_FK = " & vin & " AND BdngDate > " & dOneYear &
";"
'get record sets for the above queries
set rsVessel = OBJdbConnection.execute(VSQL)
set rsOwOp = OBJdbConnection.execute(OSQL)
set rsBoarding = OBJdbConnection.execute(BSQL)
'calculate ship type targeting numbers
strVType = rsVessel(1)
dBldDate = rsVessel(2)
Select Case strVType
          Case "TANK"
                         iShipType = 1
          Case "PASS"
                         iShipType = 1
          Case "GAS"
                         iShipType = 1
          Case "FRT"
                         'do nothing
          Case "Bulk"
                         if dBldDate < dTenYears then
                           iShipType = 2
                         End if
End Select
'Calculate owner operator targeting number
if rsOwOp.bof and rsOwOp.eof then
  errOwOp = 1
  strOwOpErr = "Owner and Operator not entered in database."
  iOwOp = 0
Else
  do while not rsOwOp.eof
    OTSQL = "SELECT TPoints FROM INTPARTY WHERE PartyID = " &
rsOwOp(0) & ";"
    rOwOp = OBJdbConnection.execute(OTSQL)
    iOwOp = iOwOp + rOwOp(0)
    rsOwOp.MoveNext
  loop
End If
'get Flag targeting number
```

```
FLSQL = "SELECT TPoints FROM FLAGSTATE WHERE FlagID = " & rsVessel(3) &
";"
rsFlag = OBJdbConnection.execute(FLSQL)
iFlag = rsFlag(0)
'get Class targeting number
If rsVessel(4) = Null then
  errClass = 1
  strClassErr = "Class not entered in database."
  iClass = 0
Else
  CLSQL = "SELECT TPoints FROM CLASS WHERE ClassID = " & rsVessel(4) & ";"
  rsClass = OBJdbConnection.execute(CLSQL)
  iClass = rsClass(0)
End If
'calculate history targeting number
if rsBoarding.bof and rsBoarding.eof then
  iHistory = 2
Else
  if rsBoarding(1) <> "NOBD" and rsBoarding(2) < dSixMonths then
         iHistory = iHistory + 1
  End if
  do while not rsBoarding.eof
    strType = rsBoarding(1)
    bDetained = rsBoarding(3)
    bOpControl = rsBoarding(4)
    Select Case strType
                Case "CAS"
                        iHistory = iHistory + 1
                Case "VIO"
                        iHistory = iHistory + 1
    End Select
    if bDetained then iHistory = iHistory + 1
    if bOpControl then iHistory = iHistory +1
         rsBoarding.MoveNext
  loop
End If
'Add up the numbers and assign priority
If errOwOp = 1 Or errClass = 1 then
  strReply = "Expect at least a call from the local unit for more information."
Else
  iTotal = iOwOp + iFlag + iClass + iHistory + iShipType
  If iTotal > 16 then
    iPriority = 1
  ElseIf iTotal > 6 then
```

```
    iPriority = 2
  ElseIf iTotal > 3 then
    iPriority = 3
  Else
    iPriority = 4
  End If
End If
'Store the points and priority of the vessel for later use
newCase = RsBoardinglist(0)
PPSQL = "UPDATE BOARDING SET TPoints= " & iTotal & ", "
PPSQL = PPSQL & "Priority=" & iPriority & " "
PPSQL = PPSQL & "WHERE CaseNum=" & newCase & ";"
OBJdbConnection.execute(PPSQL)
%>
<hr size="3" width="70%" align="left"><br>
Vessel Grading results for the vessel <%=rsVessel(0)%>.<br>
<%if errOwOp = 1 then
  Response.Write("Grading could not be completed." & strReply & " ")
  Response.Write("The information missing is: " & strOwOpErr & " ")
 ElseIf errClass = 1 then
  Response.Write("Grading could not be completed." & strReply & " ")
  Response.Write("The information missing is: " & strClassErr & " ")
 Else
   if iPriority = 4 then
         Response.Write("The vessel is a priority 4 vessel do not expect a boarding.")
   ElseIf iPriority = 3 then
         Response.Write("The vessel is a priority 3 vessel there may be a boarding.")
   ElseIf iPriority = 2 then
         Response.Write("The vessel is a priority 2 vessel expect a boarding.")
        ElseIf iPriority = 1 then
          Response.Write("The vessel is a priority 1 vessel it will be boarded.")
               Response.Write("Expect a call from the local CG unit concerning holding
the vessel out of port.")
   End If
 End If
OBJdbConnection.close
%>
<br>
</body>
<img src="images/rc_rod2.gif" width="538" height="6" border="0"><br>
<a href="home.htm">Return to Home Page</a>
</html>
```

J. FLAGDOWN.ASP

```
<%
'set up SQL query
SQLFLAG="SELECT FlagID, CountryName FROM FLAGSTATE"
set connflag = server.createobject("ADODB.Connection")
connflag.open "test2"
'get list of flags and country names from database
set flags=connflag.execute(SQLFLAG)
%>
<select name="FlagID">
                <% do while not flags.eof %>
                  <Option value = "<%= flags(0) %>"> <%= flags(1) %></Option>
                <%flags.movenext
                loop%>
</select>
<% connflag.close %>
```

K. CG_TO_REVIEW.ASP

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.0 Transitional//EN">
<html>
<head>
        <title>Vessel Cases to Review</title>
</head>
<img src="images/medbar.jpg" width="368" height="39" border="0">
<body>
<hr size="3" width="100%"><br>
To review one of the vessel boardings from the list below, click on the case <br>
number to pull up the case information.<br>
<hr size="3" width="80%" align="left"><br>
<%
port = Request.Form("PortID")
'set up SQL to get vessel recordset
SQL= "SELECT CaseNum, Type, BdngDate, Priority, Vin_FK FROM BOARDING
WHERE PortID_FK = " & port & " AND"
SQL= SQL & " InProcess = True AND Closed = False ORDER BY TPoints DESC;"
'get port name for completeness
PSQL = "SELECT UnitName FROM PORT WHERE PortID = " & port & ";"
'set up connection to the database
set conn1 = server.createobject("ADODB.Connection")
conn1.open "test2"
set rsBoarding = conn1.execute(SQL)
set rsPort = conn1.execute(PSQL)
```

```
%>
<table border>
<Caption><b><big>Vessel Boardings at  <%=rsPort(0)%></big></b></Caption>
<THEAD>
        <TR>
         <TH>Case Number</TH>
              <TH>Boarding Type</TH>
              <TH>Vessel Name</TH>
              <TH>VIN</TH>
              <TH>Flag</TH>
              <TH>Boarding Date</TH>
              <TH>Priority</TH>
        </TR>
</THEAD>
<TBODY>
<%
do while not rsBoarding.eof
VSQL= "SELECT VslName, FlagID_FK FROM VESSEL WHERE Vin = " &
rsBoarding(4) & ";"
set rsVessel = conn1.execute(VSQL)
FSQL= "SELECT CountryName FROM FLAGSTATE WHERE FlagID = " &
rsVessel(1) & ";"
set rsFlag = conn1.execute(FSQL)
%>
<tr>
    <td><a
href="cg_vsl_review_pg.asp?CaseNum=<%=rsBoarding(0)%>"><%=rsBoarding(0)%><
/a></td>
    <td><%=rsBoarding(1)%></td>
        <td><%=rsVessel(0)%></td>
        <td><%=rsBoarding(4)%></td>
        <td><%=rsFlag(0)%></td>
        <td><%=rsBoarding(2)%></td>
        <td><%=rsBoarding(3)%></td>
</tr>
<%rsBoarding.MoveNext
loop%>
</TBODY>
</table>
<br><br>
</body>
<img src="images/rc_rod2.gif" width="538" height="6" border="0"><br>
<a href="home.htm">Return to Homepage</a>
</html>
```

L. CG_VSL_REVIEW_PG.ASP

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.0 Transitional//EN">
<html>
<head>
	<title>Case Review Page</title>
</head>
<img src="images/medbar.jpg" width="368" height="39" border="0">
<body>
<hr size="3" width="100%"><br>
<%
caseNum = Request.Querystring("CaseNum")
BSQL= "SELECT * FROM BOARDING WHERE CaseNum = " & caseNum & ";"
'set up connection to the database
set conn1 = server.createobject("ADODB.Connection")
conn1.open "test2"
rsBoarding = conn1.execute(BSQL)
'get vessel information
VSQL = "SELECT * FROM VESSEL WHERE Vin = " & rsBoarding(14) & ";"
rsVessel = conn1.execute(VSQL)
FSQL = "SELECT CountryName FROM FLAGSTATE WHERE FlagID = " &
rsVessel(11) & ";"
rsFlag = conn1.execute(FSQL)
 %>
<center><big><b>Boarding Case Information</b></big></center><br>
<center><i><font color="blue">Items in blue cannot be edited.</font></i></center>
<hr size="3" width="60%"><br>
<center><form action="">
<table border>
<thead>
 <tr>
   <th>VIN</th>
	<th>Vessel Name</th>
	<th>IMO Number</th>
	<th>Flag State</th>
 </tr>
</thead>
<tr>
  <td><font color="blue"><%=rsVessel(0)%></font></td>
  <td><font color="blue"><%=rsVessel(2)%></font></td>
  <td><font color="blue"><%=rsVessel(1)%></font></td>
  <td><font color="blue"><%=rsFlag(0)%></font></td>
</tr>
```

```
</table>
<table border>
<thead>
        <tr>
                <th>Case Number</th>
                <th>Boarding Type</th>
                <th>Arrival Date</th>
                <th>Priority</th>
                <th>Boarding Date</th>
                <th>Boarding Location</th>
        </tr>
</thead>
<tr>
 <td><center><font color="blue"><%=rsBoarding(0)%></font></center></td>
 <td><input name="type" type="text" size="8" value="<%= rsBoarding(1)%>"></td>
 <td><input name="adate" type="text" size="10" value="<%=rsBoarding(2)%>"></td>
 <td><center><font color="blue"><%=rsBoarding(4)%></font></center></td>
 <td><input name="bdngdate" type="text" size="8"
value="<%=rsBoarding(5)%>"></td>
 <td><input name="bdngloc" type="text" size="15"
value="<%=rsBoarding(6)%>"></td>
</tr>
</table>
<table border>
<thead>
        <tr>
                <th>Vessel Detained?</th>
                <th>Op Control Imposed?</th>
                <th>Time on Boarding</th>
        </tr>
</thead>
<tr>
        <td><center><input type="checkbox" name="detained"
value="<%=rsBoarding(8)%>"></center></td>
        <td><center><input type="checkbox" name="opcontrol"
value="<%=rsBoarding(9)%>"></center></td>
        <td><center><input name="bdngtime" type="text" size="5"
value="<%=rsBoarding(13)%>"></center></td>
</tr>
</table>
<table border>
<thead>
        <tr>
                <th>Boarding Details</th>
```

```
		</tr>
</thead>
		<tr>
			<td>
<textarea name="detail" rows=6 cols=30><%=rsBoarding(12) %></textarea><br>
<input type="checkbox" name="validate" value="">Check here to validate case.<br>
<input type="button" value="Submit">
</td>
		</tr>
</table>
</form>
</center>
</body>
<img src="images/rc_rod2.gif" width="538" height="6" border="0"><br>
<a href="home.htm">Return to Home Page</a>
</html>
```

M. VESSEL_ENTRY.HTM

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.0 Transitional//EN">
<html>
<head>
	<title>Vessel Arrival Entry Page</title>
</head>
<body>
<img src="images/medbar.jpg" width="368" height="39" border="0" alt="CG Bar"
align="middle">
<hr size="3" width="100%">
<form action="results.asp" method="POST" name="enter">
<b>Vessel Name:</b><input name="vessel" type="text" align ="TOP"
size="30"><br><br>
    <input type="submit" value="Submit"
align="MIDDLE">  <input type="reset">
</form>
<img src="images/rc_rod2.gif" width="538" height="6" alt="" border="0"><br>
</body>
<a href="home.htm">Return to Home Page</a>
</html>
```

N. RESULTS.ASP

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.0 Transitional//EN">
<html>
```

```
<head>
	<title>Pick the Vessel</title>
</head>
<body>
<img src="images/medbar.jpg" width="368" height="39" border="0" alt="CG
Bar"><br>
<hr size="3" width="100%"><br>
<%
'declare variables
Dim vessel
Dim SQL
'set vessel equal to what user typed in
vessel=Request.Form("vessel")
'define SQL for query
SQL="SELECT Vin, IMONumber, VslName, CountryName FROM VESSEL,
FLAGSTATE "
SQL= SQL & "WHERE VslName ='" & vessel & "'"
SQL = SQL & "AND VESSEL.FlagID_FK = FLAGSTATE.FlagID"
'define SQL to find the flag name for each vsl
'create the connection
set conn = server.createobject("ADODB.Connection")
conn.open"test2"
set results=conn.Execute(SQL)
'test to see if any records found
If results.bof and results.eof then
  Response.Write("Vessel not found.")%><br>
  <a href="enter_new.asp?vslName=<%=vessel%>">Click here to enter vessel
information.</a>
<%
Else
	Response.Write("Click on the vessel id you wish to provide the arrival notice
for.")
	do while not results.eof
	%> <br>
	<table>
<tr>
    <td><a href="enter_arrival.asp?vin=<%= results(0)
%>&vslName=<%=results(2)%>&imo=<%=results(1)%>"><%= results(1)
%></a></td>
    <td><%= results(2)%></td>
    <td><%=results(3) %></td>
</tr>
  <%
	results.movenext
```

```
        loop%> <%
End if%>
</table>
<%
conn.close
 %>
</body>
<img src="images/rc_rod2.gif" width="538" height="6" border="0"><br>
<a href="home.htm">Return to Home Page</a>
</html>
```

O. ENTER_NEW.ASP

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.0 Transitional//EN">
<html>
<head>
        <title>Vessel Entry</title>
</head>
<body>
<img src="images/medbar.jpg" width="368" height="39" border="0" alt="CG
Bar"><br>
<hr size="3" width="100%"><br>
<% vessel = Request.Querystring("vslName") %>
Please enter the vessel data below:<br>
<form action="ack.asp" method="POST" name="reply">
Vessel Name:  <input name="vslName" type="text" align="TOP" size="30"
value="<%=vessel%>"><br>
IMO Number: <input name="IMO" type="text" align="TOP" size="15"><br>
Build Date:      <input name="BldDate" type="text"
align="TOP" size="15">(If not known please leave blank)
<br>
Flag:         
      <!--#include file="flagdown.asp"-->
<br>
Vessel Type:    <select name="vtype">
   <option value="TANK">Tank Ship</option>
        <option value="FRT">Freight Ship</option>
        <option value="BULK">Bulk Carrier</option>
        <option value="PASS">Passenger Ship</option>
        <option value="GAS">LPG/Gas Carrier</option>
</select>
<br>
<hr size="3" width="60%" align="left">
```

```
Arrival Date:    <input name="adate" type="text" align="TOP"
size="20"><br>
Port:
            
  
<!--#include file="portdown.asp"--><br>
Agent Info:      <!--#include file="agentdown.asp"--><br>
<br><input type="submit" value="Submit">  <input type="reset">
</form>
</body>
<img src="images/rc_rod2.gif" width="538" height="6" border="0" alt="CG Rod">
</html>
```

P. ACK.ASP

```
<!DOCTYPE html PUBLIC "-//W3C//DTD HTML 4.0 Transitional//EN">
<html>
<head>
<title>Vessel Entered and Arrival Accepted</title>
</head>
<body>
<img src="images/medbar.jpg" width="368" height="39" border="0" alt="CG
Bar"><br>
<hr size="3" width="100%">
<br>
<%
IMO = Request.Form("IMO")
vslName = Request.Form("vslName")
FlagID = Request.Form("FlagID")
vtype = Request.Form("vtype")
BldDate = CDate(Request.Form("BldDate"))
vlen =1
bre = 1
depth = 1
prop = "unkn"
snote = "none"
'setup Insert query to create the vessel in the database
SQLINSERT = "INSERT INTO VESSEL (IMONumber, VslName, VslType, BldDate,
Length, Breadth, Depth, Propulsion, SpecialNote, FlagID_FK) "
SQLINSERT = SQLINSERT &mp; "VALUES ("
SQLINSERT = SQLINSERT &mp; IMO &mp; ", "
SQLINSERT = SQLINSERT &mp; "'" &mp; vslName &mp; "', "
SQLINSERT = SQLINSERT &mp; "'" &mp; vtype &mp; "', "
SQLINSERT = SQLINSERT &mp; "'" &mp; BldDate &mp; "', "
```

```
SQLINSERT = SQLINSERT &mp; vlen &mp; ", "
SQLINSERT = SQLINSERT &mp; bre &mp; ", "
SQLINSERT = SQLINSERT &mp; depth &mp; ", "
SQLINSERT = SQLINSERT &mp; "'" &mp; prop &mp; "', "
SQLINSERT = SQLINSERT &mp; "'" &mp; snote &mp; "', "
SQLINSERT = SQLINSERT &mp; FlagID &mp; "); "  %> <%
'create the vessel record
set conninsert = server.createobject("ADODB.Connection")
conninsert.open "test2"
conninsert.execute(SQLINSERT)
'get vin for vessel just entered
SQLV = "SELECT Vin FROM VESSEL WHERE VslName ='" &mp; vslName &mp;"'"
set rVin = conninsert.execute(SQLV)
'set up query to record vessel arrival information
adate = CDate(Request.Form("adate"))
port = Request.Form("PortID")
btype = "NOBD"
agent = Request.Form("AgentID")
Vin = rVin(0)
SQLBD = "INSERT INTO BOARDING (Type, ArrivalDate, Vin_FK, PortID_FK,
PartyID_FK) "
SQLBD = SQLBD &mp; "VALUES ("
SQLBD = SQLBD &mp; "'" &mp; btype &mp; "', "
SQLBD = SQLBD &mp; "'" &mp; adate &mp; "', "
SQLBD = SQLBD &mp; "'" &mp; Vin &mp; "', "
SQLBD = SQLBD &mp; "'" &mp; port &mp; "', "
SQLBD = SQLBD &mp; "'" &mp; agent &mp; "') "
conninsert.execute(SQLBD)
'get the port name for acknowledgement
SQLP = "SELECT UnitName FROM PORT WHERE PortID = " &mp; port &mp; ";"
set rName = conninsert.execute(SQLP)
pName = rName(0)
conninsert.close
%> <br>
Arrival accepted for the <%= vslName %> in <%=pName%> arriving on
<%=adate%>.<br>
<br>
<a href="grading_results.asp?vin=&lt;%=Vin%&gt;">Click here to view the grading
results on the vessel.</a><img src="images/rc_rod2.gif" width="538" height="6"
border="0">
</body>
</html>
```

Q. GRADING_RESULTS.ASP

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.0 Transitional//EN">
<html>
<head>
	<title>Vessel Grading Results</title>
</head>
<img src="images/medbar.jpg" width="368" height="39" border="0"><br>
<hr size="3" width="100%"><br>
<body>
<%
'declare targeting variables
dim iOwOp, iFlag, iClass, iHistory, iShipType, iTotal, iPriority
'setup variables for use in targeting
vin = Request.Querystring("vin")
dToday = Date
dOneYear = DateAdd("yyyy", -1, dToday) 'get date for a year ago
dTenYears = DateAdd("yyyy", -10, dToday) 'get a date for ten years ago
dSixMonths = DateAdd("m", -6, dToday) 'get a date six months ago
'setup query to get vessel info
VSQL = "SELECT VslName, VslType, BldDate, FlagID_FK, ClassID_FK FROM "
VSQL = VSQL & "VESSEL WHERE Vin = " & vin & ";"
'setup query to get owner and operator for vessel
OSQL = "SELECT PartyID_FK FROM [VESSEL-INTPARTY] WHERE Vin_FK = " &
vin & ";"
'setup query to get boardings for the vessel
BSQL = "SELECT CaseNum, Type, BdngDate, VslDetained, OpControl FROM
BOARDING "
BSQL = BSQL & "WHERE Vin_FK = " & vin & " AND BdngDate > " & dOneYear &
";"
'create connection to the database
set conn1 = server.createobject("ADODB.Connection")
conn1.open "test2"
'get record sets for the above queries
set rsVessel = conn1.execute(VSQL)
set rsOwOp = conn1.execute(OSQL)
set rsBoarding = conn1.execute(BSQL)
'calculate ship type targeting numbers
strVType = rsVessel(1)
dBldDate = rsVessel(2)
Select Case strVType
		Case "TANK"
				iShipType = 1
		Case "PASS"
```

```
                iShipType = 1
        Case "GAS"
                iShipType = 1
        Case "FRT"
                'do nothing
        Case "Bulk"
                if dBldDate < dTenYears then
                  iShipType = 2
                End if
End Select
'Calculate owner operator targeting number
if rsOwOp.bof and rsOwOp.eof then
  errOwOp = 1
  strOwOpErr = "Owner and Operator not entered in database."
  iOwOp = 0
Else
  do while not rsOwOp.eof
    OTSQL = "SELECT TPoints FROM INTPARTY WHERE PartyID = " &
rsOwOp(0) & ";"
    rOwOp = conn1.execute(OTSQL)
    iOwOp = iOwOp + rOwOp(0)
    rsOwOp.MoveNext
  loop
End If
'get Flag targeting number
FLSQL = "SELECT TPoints FROM FLAGSTATE WHERE FlagID = " & rsVessel(3) &
";"
rsFlag = conn1.execute(FLSQL)
iFlag = rsFlag(0)
'get Class targeting number
If rsVessel(4) = Null then
  errClass = 1
  strClassErr = "Class not entered in database."
  iClass = 0
Else
  CLSQL = "SELECT TPoints FROM CLASS WHERE ClassID = " & rsVessel(4) & ";"
  rsClass = conn1.execute(CLSQL)
  iClass = rsClass(0)
End If
'calculate history targeting number
if rsBoarding.bof and rsBoarding.eof then
  iHistory = 2
Else
  if rsBoarding(1) <> "NOBD" and rsBoarding(2) < dSixMonths then
```

```
        iHistory = iHistory + 1
  End if
  do while not rsBoarding.eof
    strType = rsBoarding(1)
    bDetained = rsBoarding(3)
    bOpControl = rsBoarding(4)
    Select Case strType
                    Case "CAS"
                            iHistory = iHistory + 1
                    Case "VIO"
                            iHistory = iHistory + 1
    End Select
    if bDetained then iHistory = iHistory + 1
    if bOpControl then iHistory = iHistory +1
        rsBoarding.MoveNext
  loop
End If
'Add up the numbers and assign priority
If errOwOp = 1 Or errClass = 1 then
  strReply = "Expect at least a call from the local unit for more information."
Else
  iTotal = iOwOp + iFlag + iClass + iHistory + iShipType
  If iTotal > 16 then
    iPriority = 1
  ElseIf iTotal > 6 then
    iPriority = 2
  ElseIf iTotal > 3 then
    iPriority = 3
  Else
    iPriority = 4
  End If
End If
%>
Vessel Grading results for the vessel <%=rsVessel(0)%>.<br>
<%if errOwOp = 1 then
  Response.Write("Grading could not be completed." & strReply & " ")
  Response.Write("The information missing is: " & strOwOpErr & " ")
 ElseIf errClass = 1 then
  Response.Write("Grading could not be completed." & strReply & " ")
  Response.Write("The information missing is: " & strClassErr & " ")
 Else
   if iPriority = 4 then
          Response.Write("The vessel is a priority 4 vessel do not expect a boarding.")
   ElseIf iPriority = 3 then
```

```
        Response.Write("The vessel is a priority 3 vessel there may be a boarding.")
   ElseIf iPriority = 2 then
        Response.Write("The vessel is a priority 2 vessel expect a boarding.")
      ElseIf iPriority = 1 then
        Response.Write("The vessel is a priority 1 vessel it will be boarded.")
            Response.Write("Expect a call from the local CG unit concerning holding
the vessel out of port.")
   End If
 End If
conn1.close
%>
</body>
<img src="images/rc_rod2.gif" width="538" height="6" border="0">
</html>
```

R. CG_NO_BOARDS.ASP

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.0 Transitional//EN">
<html>
<head>
        <title>Vessels in Port</title>
</head>
<img src="images/medbar.jpg" width="368" height="39" border="0">
<body>
<hr size="3" width="100%"><br>
<%
port = Request.Form("PortID")
'set up SQL to get vessel recordset
SQL= "SELECT CaseNum, ArrivalDate, DateClosed, Vin_FK, PartyID_FK FROM
BOARDING WHERE PortID_FK = " & port & " AND"
SQL= SQL & " Type = 'NOBD' AND Closed = False ORDER BY ArrivalDate;"
'get port name for completeness
PSQL = "SELECT UnitName FROM PORT WHERE PortID = " & port & ";"
'set up connection to the database
set conn1 = server.createobject("ADODB.Connection")
conn1.open "test2"
set rsBoarding = conn1.execute(SQL)
set rsPort = conn1.execute(PSQL)
%>
This is the list of vessels in the port that were not boarded. Select the vessels which
have<br>
departed by checking the check box and enter the departure date in the departure date
field.<br><br>
<hr size="3" width="70%" align="left">
```

When finished, click on the "submit" button.

```
<form action="cg_nobd_confirmation.asp" method="POST" name="list">
<table border>
<Caption><b><big>List of Not Boarded Vessels in Port for
<%=rsPort(0)%></big></b></Caption>
<THEAD>
        <TR>
         <TH>Chk</TH>
              <TH>Vessel Name</TH>
              <TH>VIN</TH>
              <TH>Flag</TH>
              <TH>Arrival Date</TH>
              <TH>Agent Name</TH>
              <TH>Date Departed</TH>
        </TR>
</THEAD>
<TBODY>
<%
do while not rsBoarding.eof
VSQL= "SELECT VslName, FlagID_FK FROM VESSEL WHERE Vin = " &
rsBoarding(3) & ";"
set rsVessel = conn1.execute(VSQL)
FSQL= "SELECT CountryName FROM FLAGSTATE WHERE FlagID = " &
rsVessel(1) & ";"
set rsFlag = conn1.execute(FSQL)
ASQL= "SELECT PartyName FROM INTPARTY WHERE PartyID = " &
rsBoarding(4) & ";"
set rsAgent = conn1.execute(ASQL)
%>
<tr>
    <td><input type="checkbox" name="box" value="<%=rsBoarding(0)%>"></td>
    <td><%=rsVessel(0)%></td>
        <td><%=rsBoarding(3)%></td>
        <td><%=rsFlag(0)%></td>
        <td><%=rsBoarding(1)%></td>
        <td><%=rsAgent(0)%></td>
        <td><input name="close" type="text" value="<%=Date%>"></td>
</tr>
<%rsBoarding.MoveNext
loop%>
</TBODY>
</table><br>
<input type="submit" value="Submit"> <input type="reset">
</form>
```

```
</body>
<img src="images/rc_rod2.gif" width="538" height="6" border="0"><br>
<a href="home.htm">Return to Home Page</a>
</html>
```

S. CG_NOBD_CONFIRMATION.ASP

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.0 Transitional//EN">
<html>
<head>
	<title>Confirmation of Vessel Departure</title>
</head>
<img src="images/medbar.jpg" width="368" height="39" border="0">
<body>
<hr size="3" width="100%"><br>
<%
'set up connection to the database
set conn1 = server.createobject("ADODB.Connection")
conn1.open "test2"
'loop through the checkboxes passed from the last page
For Each item In Request.Form("box")
 'update record to indicate vessel is selected for a boarding
 USQL = "UPDATE BOARDING SET Closed = True, DateClosed = '" &
Request.Form("close") & "' WHERE CaseNum = " & item & ";"
 conn1.execute(USQL)
Next
%>
Vessel(s) selected for departure updated.
<br>
</body>
<img src="images/rc_rod2.gif" width="538" height="6" border="0"><br>
<a href="home.htm">Return to Home Page</a>
</html>
```

LIST OF REFERENCES

Baden, Kurt A. and Peters Gerald A., *A Business Process Model and Reengineering Plan for the Student Services Department of the Marine Corps Institute*, Master's Thesis, Naval Postgraduate School, Monterey, California, September 1997.

BreezeCom, "Solutions", [http://www.breezecom.com/Solutions/overview.htm], May 2000.

Caudle, Sharon L., *Reengineering for Results: Update, National Academy of Public Administration Foundation*, Washington, D.C., 1995

Davenport, T. H. and Short, J. E., "The New Industrial Engineering: Information Technology and Business Process Redesign", *Sloan Management Review*, Summer 1990.

Hammer, Michael, "Reengineering Work: Don't Automate, Obliterate", *Harvard Business Review*, July – August 1990.

Hammer, Michael and Champy, James, *Reengineering the Corporation*, Nicholas Brealey Publishing Ltd, 1993.

Hoffer, Jeffery A., George, Joey F., and Valacich, Joesph S., *Modern Systems Analysis & Design*, Second Edition, Addison Wesley Longman, Inc., 1999.

KOPeR, "KOPeR Redesign Agent", [http://joshua.nps.navy.mil:8080/koper/form.htm], March 2000.

Kroenke, David M., *Databse Processing: Fundamentals, Design, and Implementation*, Sixth Edition, Prentice Hall, Inc., 1998.

Microsoft Press, *Microsoft Internet Information Server*, 1998.

Nissen, Mark E., "Redesigning Reengineering Through Measurement – Driven Inference", *MIS Quarterly*, December 1998, pp. 509-534.

Orinoco Inc, "Frequently Asked Questions", [http://www.wavelan.com/products/faq/faq.html?FAQ_id=11], May 2000.

United States Coast Guard, *Information Technology Management Strategy*, Version 1.0, USCG Director of Information and Technology, 1998.

United States Department of Transportation, *U. S. Coast Guard Marine Safety Manual*, Volume II, Government Printing Office, Washington, D.C., 1996

THIS PAGE INTENTIONALLY LEFT BLANK

INITIAL DISTRIBUTION LIST

1. Defense Technical Information Center 2
   8725 John J. Kingman Road, Ste 0944
   Fort Belvoir, VA 22060-6218

2. Dudley Knox Library 2
   Naval Postgraduate School
   411 Dyer Road
   Monterey, California 93943-5101

3. Professor Daniel Dolk (Code IS/DK) 1
   Naval Postgraduate School
   Monterey, CA 93943

4. Professor Mark Nissen (Code SM/NI) 1
   Naval Postgraduate School
   Monterey, CA 93943

5. Commandant (G-SRF) 1
   2100 Second Street SW
   Washington, DC 20593-0001
   Attn: LCDR Jan Stevens

6. LCDR Jason Fosdick. 1
   5743 Moonbeam Drive
   Woodbridge, VA 22193